NASA Technical Memorandum 108514

Computer-Aided System Engineering and Analysis (CASE/A) User's Manual Version 5.0

J.C. Knox, Editor

June 1996

NASA Technical Memorandum 108514

Computer-Aided System Engineering and Analysis (CASE/A) User's Manual Version 5.0

J.C. Knox, Editor
Marshall Space Flight Center • MSFC, Alabama

National Aeronautics and Space Administration
Marshall Space Flight Center • MSFC, Alabama 35812

June 1996

VERSION 5.0 RELEASE NOTES

• WARNING ABOUT INSTALLING NEW CODE—Please communicate with the systems manager during the installation process of the new code. **BEWARE: If the component data bases are replaced with the new data bases, this will destroy the performance data of all existing models that were previously built.**

To avoid this:

(1) Copy only the source code from tape to the [CASEA.CODE] directory.

(2) Then copy your component data bases to the [CASEA.DATA] directory.

If, on the other hand, your version of CASE/A is below 4.0 and/or you have only used the package from a learning perspective (i.e., no models of importance have been developed), you may consider replacing the component data base directory [CASEA.DATA] to have a clean installation.

<u>If currently using CASE/A version 4.0, please be aware of the following changes</u>:

- LOGICALS—The logical assignments used in CASE/A have been changed to avoid potential conflicts with VAX system logicals on your system. These changes must be incorporated into your login.com file to ensure proper operation of version 5.0. Further explanation is provided in appendix C.

  New assignments:

  $ ASSIGN *DEVICE*:[CASEA.CODE] CASEA$CODE
  $ ASSIGN *DEVICE*:[CASEA.DATA] CASEA$DATA

  where *DEVICE* is the name of your local VAX disk drive (e.g. DUA1, $DISK1, etc.).

- MODULE—This simulation component has been added in version 5.0 to assist engineers in the analysis of space station life support system designs. The MODULE component is made up of three discrete volumes (Rack, Aisleway, and Standoff) of which the Rack volume may enclose other equipment/components. The ASSIGN command associates these components to the enclosure. During the simulation, the Module component tracks the respirable atmospheric composition, mass additions/losses, and heat transfer to and from other modules, enclosed equipment and the external orbital environment. As with other components, Module simulation output data can be captured and analyzed using the integrated plot utility. Please refer to section 10—Module Component for a technical description of this new component.

- ARCHIVE UTILITY—The archive utility has been added in version 5.0 to provide the user with more flexibility in managing multiple CASE/A simulation solutions. This utility gives the user the ability to archive a complete simulation solution including user-defined OPS code, output plots, and all other relevant data to allow reconstruction of the model at a later time. One benefit is that the user, after setting up a model, can choose to archive that model for later recall before making similar solution runs with variants of that particular model. Hence, variants of the model can be readily compared. When the user wishes to reexamine the archived model, they simply enter the RETRIEVE command to reconstruct the model for execution and/or examination. Please refer to section 6 for more information about the archive utility.

- MERGE COMMAND—The merge command has been enhanced in version 5.0. The user now has the ability to merge cases from different CASE/A versions. The merge utility supports the user in

developing large and complicated models. Complete subsystems are constructed and verified as small, manageable cases, then merged into a larger model. Development time can be reduced by dealing with smaller, more manageable parts of the system. Please refer to section 6 for more information about the merge utility.

- AUTOPLOT UTILITY—The autoplot utility has been added in version 5.0 to provide the user with the capability to automatically generate the set of all plots for a specified model. When invoked this utility generates the set of all plots using the parameters defined to be output as the model simulation is run. Plot files can be generated either interactively or in batch. For a large number of plots, the batch method is more efficient. Please refer to section 5 for more information about the autoplot utility.

- SAMPLE PROBLEMS—Updates to the sample problems in version 5.0 allow the new user to get started using CASE/A in less time. Sample problem 1, the active thermal control (ATC) system model was modified to provide the new user with an easy method to get started using CASE/A. Sample problem 2, Mission Build 6 (MB6) is new to version 5.0 and demonstrates the capabilities in CASE/A for managing large models such as this one (capabilities such as archiving/retrieving models and organizing the schematic for large models). The third sample problem in appendix A (Water Conditioning Model (WCM)) has been enhanced to include several more output plots to assist the user in developing this more complex model.

- CASEA$TDMS—This logical, which pointed to the TDMS directory, is no longer used. All routines which manage the component data bases have been moved from the TDMS directory to the CODE directory, so that now, the logical assignment CASEA$CODE (as described above) is used within the CASE/A code to access these routines.

If you do not have CASE/A installed on your system, please note:

- INSTALLING NEW CODE—Be sure to communicate with your systems manager during the installation process of the new code. A single sheet of installation instructions has been provided with the tape. If additional information is required, please consult appendix C.

PLEASE NOTE: If the user finds that the model is not converging for cases with a large number of components, please try the “Feedback” solution method (specified in the CONTROL file). In the past, problems with convergence have been noted where the “Matrix” solution method was used for large cases.

VERSION 4.1 RELEASE NOTES

- CASEA.DAT—This file, which pointed to the component data base directory, is no longer used. The pointer is replaced with the logical assignment CASEA$DATA (as described above).

- BLACKBOX—This feature has been enhanced in version 4.1. CASE/A now supports up to 25 different BLACKBOX components. Subroutine OPS7 has been modified to support this enhancement. If you are currently using any BLACKBOX components, be sure to read sections 7.6.3 and 10.4 for additional information.

- SAMPLE PROBLEM 1—The first sample problem in appendix A has been updated.

COMMANDS AND DOCUMENT FORMATTING CONVENTIONS

The following conventions are followed in the CASE/A user's manual to assist the reader:

- CASE/A commands such as "Loadcase" and its parameters are written in uppercase courier 12 font. Example: `LOADCASE MB6_1`
- VAX VMS - DCL commands are also written in uppercase courier 12 font. Example: `$ @LINKCAS`
- Dialogs showing CASE/A sessions are shown in courier font with user responses shown in **bold.** Example: `CASEA_ATC >` **`DIR`**

CONTRIBUTORS

The original CASE/A User's Manual, Programmer's Manual, and program were created by Robert E. Ferguson, Michal E. Bangham, J. L. (Louie) Clayton, Scott D. Gilley, Roy G. Davis, Jr., Allen S. Bacskay, Brian Key, Tim C. Tripp, Robert C. DaLee, Thomas C. Lee, Terry W. Carroll, David A. Till, and Dave Anderson of the Huntsville Division of McDonnell Douglas Space Systems Company for Marshall Space Flight Center under contract NAS8-36407.

Revisions to the CASE/A program have been made by Scott D. Gilley and Sam Edwards of the MSFC Group of Sverdrup Technology for Marshall Space Flight Center under contract NAS8-37814, and by Harold E. Quinn and Lesa Naidenko of the Hunstville Division of Computer Sciences Corporation for Marshall Space Flight Center under contract NAS8-60000.

Revisions to the CASE/A User's Manual and Programmer's Manual have been made by Harold E. Quinn and Lesa Naidenko of the Hunstville Division of Computer Sciences Corporation for Marshall Space Flight Center under contract NAS8-60000.

James C. Knox of Marshall Space Flight Center has overall responsibility for changes to and maintenance of the CASE/A manuals and program.

TABLE OF CONTENTS

		Page
1.0	INTRODUCTION	1
	1.1 Basic Concepts and Development History	1
	1.2 Program Overview and Capabilities	2
	1.3 Introductory Example Models	6
	1.4 Manual Organization	17
	1.5 References	20
2.0	SCHEMATIC MANAGEMENT	21
	2.1 Program Initiation and Termination	21
	2.2 Model Creation, Loading, and Deletion	22
	2.3 Schematic Manipulation	22
3.0	COMPONENT DATA BASE MANAGEMENT	29
	3.1 Component Data Base Layout	29
	3.2 Data Base Screen Editor	29
	3.3 Data Base Line Editor	34
4.0	SIMULATION CONTROL AND EXECUTION	36
	4.1 Simulation Control Constants	36
	4.2 Chemical Constituents	37
	4.3 Plot Data Setup	38
	4.4 Canned Plotset Setup	40
	4.5 Simulation Initiation	42
	4.6 Simulation Status and Error Messages	43
5.0	MODEL OUTPUT MANAGEMENT	44
	5.1 Schematic Output	44
	5.2 Component Data Base Output	45
	5.3 Simulation Tabular Summary File	45
	5.4 Integrated Plot Utility	47
	5.5 Custom User Output	55
	5.6 Schematic Connection and Hydraulic Maps	55
	5.7 The Autoplot Process	58
6.0	UTILITY COMMANDS	59
	6.1 Terminal Settings	59
	6.2 On-Line Help Information	59
	6.3 Temporary Exit to VAX Operating System	59
	6.4 Temporary Exit to VAX Editor	69
	6.5 Subsystem Heat Load Assignment to Cabins/Modules	60
	6.6 Merge Operation Across VAX Hosts	60
	6.7 Archive Setup Process	61
	6.8 Archive Process	63
	6.9 Retrieve Archive Model	63

TABLE OF CONTENTS (Continued)

Page

7.0 USER OPERATIONS LOGIC 65

7.1 OPS Logic Structure 65
7.2 System Data Storage and Manipulation 66
7.3 Component Solution Order Sequencing 72
7.4 OPS Logic Incorporation 73
7.5 OPS Logic Preprocessor 74
7.6 Special CASE/A Components 75

8.0 ANALYTICAL TECHNIQUES 77

8.1 Iterative Solution Process 77
8.2 Pressure Drop Solution 78
8.3 Heat Transfer Networks 83
8.4 Effective Wall Radiation 87
8.5 Mass Transfer Networks 87
8.6 Fluid Properties Calculation 88
8.7 References 88

9.0 USER TIPS 89

9.1 Things to Avoid 89
9.2 Hydraulic Solution Notes 90
9.3 Solution Convergence Notes 91
9.4 Modeling Suggestions 93

10.0 COMPONENT LIBRARY DESCRIPTIONS 95

10.1 ADSORPTN—Adsorption Bed Simulation Description 98
10.2 AFSPE—Anode Feed Solid Polymer Electrolysis Description 101
10.3 ALHX—Air/Liquid Heat Exchanger Simulation Description 107
10.4 BLACKBOX—BLACKBOX Simulation Description 109
10.5 BMR—Body Mounted Radiator Simulation Description 111
10.6 BOSCH—BOSCH Simulation Description 113
10.7 BUBBLE—Interconnect BUBBLE Simulation Description 119
10.8 CABIN—CABIN Simulation Description 120
10.9 CAP—Thermal Capacitor Simulation Description 123
10.10 CFR—Carbon Formation Reactor Simulation Description 125
10.11 CHX—Condensing Heat Exchanger Simulation Description 127
10.12 CNHX—Contact Heat Exchanger Simulation Description 129
10.13 CNTRLLR—Controller Simulation Description 131
10.14 CP—Coldplate Simulation Description 135
10.15 CREW—CREW Simulation Description 137
10.16 DEFLOW—Dehumidifier (Flow Desorb) Simulation Description 139
10.17 DEHUM—Dehumidifier (Vacuum Desorb) Simulation Description 141
10.18 EDC—Electrochemical Depolarized CO_2 Concentrator Simulation Description 143
10.19 EVAP—Liquid Flash Evaporator Simulation Description 146
10.20 FILTER—Filter Simulation Description 149

TABLE OF CONTENTS (Continued)

Page

10.21 H2OSEP—Water Separator Simulation Description 152
10.22 HATCH—HATCH Simulation Description..... 153
10.23 HEATER—HEATER Simulation Description 154
10.24 IONEXCH—Ion Exchange Resin Simulation Description 155
10.25 LIOH—Lithium Hydroxide Simulation Description..... 156
10.26 LLHX—Liquid/Liquid Heat Exchanger Simulation Description..... 160
10.27 MODULE—MODULE Simulation Description..... 162
10.28 MOLSIEV—Molecular Sieve Simulation Description..... 165
10.29 MSPLIT—Multisplit Simulation Description..... 171
10.30 NODE—NODE Simulation Description..... 173
10.31 O2N2—Oxygen/Nitrogen Panel Simulation Description 176
10.32 PIPE—PIPE Simulation Description..... 177
10.33 PREWAST—Waste Pretreatment Simulation Description..... 179
10.34 PUMP—PUMP Simulation Description..... 181
10.35 RACK—RACK Simulation Description..... 185
10.36 RAD—Radiator Simulation Description 187
10.37 RO—Reverse Osmosis Simulation Description..... 189
10.38 SABAT—SABAT Simulation Description 192
10.39 SAWD—Solid Amine Water Desorbed Simulation Description 200
10.40 SFWE—Static Feed Water Electrolysis Simulation Description 208
10.41 SINK—SINK Simulation Description..... 214
10.42 SOURCE—SOURCE Simulation Description 215
10.43 SPLIT—Flow SPLIT Simulation Description 216
10.44 STORE—Storage Tank Simulation Description 217
10.45 SUM—Flow Summation Simulation Description 221
10.46 TBUS—Thermal Bus Simulation Description 223
10.47 TIMER—Timer Simulation Description 224
10.48 TIMES—Thermoelectric Integrated Membrane Evaporation Subsystem Simulation Description 226
10.49 VALVE—VALVE Simulation Description 229
10.50 VCD—Vapor Compression Distillation Simulation Description 233
10.51 WASH—WASH Process Simulation Description..... 236
10.52 WQM—Water Quality Monitor Simulation Description..... 239

APPENDIX A – SAMPLE PROBLEMS..... 241

APPENDIX B – CASE/A COMMAND SUMMARY 273

APPENDIX C – CASE/A INSTALLATION NOTES..... 303

LIST OF ILLUSTRATIONS

Figure	Title	Page
1.	An overview of the CASE/A program	3
2.	CASE/A model construction process	5
3.	CASE/A file structure	6
4.	Sample problem 1: ATC subsystem schematic	7
5.	Typical component icon	8
6.	Archive edit screen for case MB6	9
7.	Archive data base screen for case MB6 and archive set MB6TEST	10
8.	Introductory example retrieve process	11
9.	Example DIR command response	14
10.	Example model schematic of subsystem LAYOUT	15
11.	Example component edit screen-menu	15
12.	Example control edit screen	16
13.	Introductory example problem partial .LPP file	17
14.	MB6_1 example CASE/A output plot from 1-h run	18
15.	MB6_1 example CASE/A output plot from 48-h run	19
16.	CON array layout	30
17.	Typical component main edit screen input area	30
18.	Typical component constituent edit screen	32
19.	Typical component main edit screen output area	33
20.	Typical component main edit screen benchmark area	34
21.	Example of component line editor	35
22.	CONTROL default edit screen	37
23.	LABELS default edit screen	38
24.	Example plot edit screen	39
25.	Example plotset definition screen	40

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
26.	Example plotset definition screen for a canned plotset—first screen	41
27.	Example plotset definition screen for a canned plotset—second screen	41
28.	Example autopilot edit screen	42
29.	Typical simulation progress display	43
30.	An example CASENAME.LPP file	46
31.	An example .CMP file	56
32.	An example .FMP file	57
33.	Example archive edit screen	62
34.	Example archive data base screen	62
35.	Example archive process	64
36.	OPS routine calls in solution flow	65
37.	CON array layout	66
38.	User constants default edit screen	71
39.	OPS logic incorporation procedure	73
40.	Example TIMER connection	75
41.	Example CNTRLLR connection	75
42.	Modified subroutine OPS7	76
43.	Component logic functions	78
44.	Four-pole noncondensing HX (SINDA analogy)	84
45.	Delta network (SINDA analogy)	84
46.	Thermal network solved in TNETWK	86
47.	Example flow generator placement schematics	91
48.	Example boundary pressure placement schematics	92

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
49.	Single cell thermal network	103
50.	Divider plate node next to H2/H2O separator	103
51.	Divider plate node next to O2/H2O separator	104
52.	BMR thermal network	112
53.	Sample logic for modifying BMR fin efficiency and sink temperature	112
54.	BOSCH thermal network	114
55.	CABIN thermal network	121
56.	CAP thermal network	124
57.	Carbon formation reactor thermal network	126
58.	CNHX thermal network	129
59.	CP thermal network	136
60.	DEHUM thermal/vapor density networks	142
61.	EDC thermal network	144
62.	EVAP thermal network	148
63.	LIOH vapor density network	157
64.	MODULE thermal network	163
65.	MOLSIEV mass transfer network	166
66.	MOLSIEV thermal network	168
67.	RACK thermal network	185
68.	RAD thermal network	188
69.	SABAT subsystem schematic	193
70.	Sabatier reactor thermal network schematic	196
71.	SAWD mass transfer network	201
72.	SAWD thermal network	205
73.	SFWE thermal network schematic	210

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
74.	Storage tank thermal network	219
75.	Percent solids versus fraction of water in feed distilled (example only)	226
76.	TIMES heat transfer network	228
77.	VCD heat transfer network	234

LIST OF TABLES

Table	Title	Page
1.	CASE/A program capabilities	6
2.	Main screen editing command functions	31
3.	Subordinate screen editing command functions	33
4.	Plotset subscreen editing command functions	33
5.	Plotset OPS activation codes	39
6.	Plotset definition data summary	40
7.	CON array data access routines	67
8.	Fluid constituents definition	68
9.	C array data access routines	68
10.	Fluid properties definition	69
11.	PRO array data access routines	69
12.	Internal simulation control variables	70
13.	User constants data access routines	72
14.	PCS interface routines	72
15.	Format guidelines for the PREPRO routine	74
16.	Fluid stream classifications	80
17.	Available equipment types	96
18.	ADSORPTN main edit screen	100
19.	ADSORPTN constituent edit screen	100
20.	AFSPE main edit screen	106
21.	AFSPE output edit screen	106
22.	ALHX edit screen	108
23.	BLACKBOX edit screen	110

LIST OF TABLES (Continued)

Table	Title	Page
24.	BMR edit screen	111
25.	BOSCH edit screen	118
26.	CABIN edit screen	122
27.	CAP edit screen	124
28.	CFR edit screen	126
29.	CHX edit screen	128
30.	CNHX edit screen	130
31.	CNTRLLR data base parameters (example only)	133
32.	Available CNTRLLR commands	133
33.	CP edit screen	136
34.	CREW edit screen	137
35.	CREW constituent edit screen	138
36.	DEFLOW edit screen	140
37.	DEFLOW vacuum desorb edit screen	140
38.	DEFLOW outputs/benchmarks edit screen	140
39.	DEHUM edit screen	142
40.	EDC edit screen	145
41.	EVAP edit screen	148
42.	FILTER edit screen	151
43.	FILTER constituent edit screen	151
44.	H2OSEP edit screen	152
45.	HATCH edit screen	153
46.	HEATER edit screen	154
47.	IONEXCH edit screen	155
48.	IONEXCH constituent edit screen	155

LIST OF TABLES (Continued)

Table	Title	Page
49.	LIOH edit screen	159
50.	LLHX edit screen	161
51.	MODULE edit screen	164
52.	MOLSIEV main edit screen	170
53.	MOLSIEV output edit screen	170
54.	MSPLIT edit screen	172
55.	NODE edit screen	175
56.	O2N2 edit screen	176
57.	PIPE edit screen	178
58.	PREWAST edit screen	180
59.	PUMP edit screen	184
60.	RACK edit screen	186
61.	RAD edit screen	188
62.	RO edit screen	191
63.	RO constituents edit screen	191
64.	SABAT default edit screen No. 1	198
65.	SABAT default edit screen No. 2	199
66.	SAWD edit screen	206
67.	SFWE curve fit edit screen	209
68.	SFWE main edit screen	213
69.	SFWE outputs edit screen	213
70.	SINK edit screen	214
71.	SOURCE main edit screen	215
72.	SOURCE constituent edit screen	215
73.	SPLIT edit screen	216

LIST OF TABLES (Continued)

Table	Title	Page
74.	STORE main edit screen	220
75.	STORE constituents edit screen	220
76.	SUM edit screen	222
77.	TBUS edit screen	223
78.	TIMER edit screen	225
79.	TIMES edit screen	227
80.	TIMES constituent edit screen	227
81.	VALVE edit screen	232
82.	VCD edit screen	234
83.	VCD constituent edit screen	235
84.	WASH edit screen	237
85.	WASH constituents edit screen	237
86.	WQM edit screen	238
87.	WQM constituents edit screen	239

TECHNICAL MEMORANDUM

COMPUTER-AIDED SYSTEM ENGINEERING AND ANALYSIS USER'S MANUAL, VERSION 5.0

1.0 INTRODUCTION

The computer aided system engineering and analysis (CASE/A) modeling package is an environmental control and life support system (ECLSS) and active thermal control system (ATCS) analysis and trade study tool. CASE/A is a graphics-based command driven package developed for the DEC VAX™ computer series using a Tektronix 4014™ or compatible for graphics, and a DEC VT100™ or compatible for text manipulation. The CASE/A system can be used productively with Tektronix graphics capabilities alone; however, the DEC VT™ series capability provides the most efficient user interface for full screen editing. Many inexpensive composite terminals and personal computer (PC) emulations offer both capabilities. The program provides a user-friendly environment in which the engineer can analyze the performance and interface characteristics of an ECLS/ATC system. CASE/A proves extremely useful during all phases of a spacecraft design program, from the initial conceptual design trade studies to the actual flight, including preflight prediction and in-flight anomaly analysis.

Minimal engineer training time (less than a month) is required to learn the basic CASE/A program. It is highly advisable for the first-time CASE/A user to read this manual thoroughly before attempting to run a model. With this knowledge, the user can construct and run a simple model in a matter of hours and obtain meaningful results. However, to take full advantage of the many features CASE/A has to offer, the advanced user should have a strong thermodynamic, heat and mass transfer, and chemical background, with a working knowledge of the FORTRAN programming language. The educational background is required to fully understand the analytical techniques used to model the component routines that are at the heart of a CASE/A simulation. The FORTRAN knowledge is required for the user to interact with the simulation through operations logic and/or through custom designed component routines.

1.1 Basic Concepts and Development History

The CASE/A program consists primarily of an interactive graphics-based system schematic generator, a data base management system, and a library of component subroutines. The program operates on user-defined system schematics based on a library of components resident within the program. The program translates these schematics into the required internal component interconnect and data definition arrays to support the solution routine. CASE/A provides for transient tracking of the flow stream constituents and determination of their thermodynamic state throughout an ECLSS/ATCS simulation. The program tracks each constituent through all combination and decomposition states while maintaining both a mass and energy balance on the overall system. This allows rapid assessment of ECLSS designs, the impact of alternate technologies, and impacts due to changes in metabolic forcing functions, consumables usage, and system control considerations. Changes to a model can be rapidly performed interactively by manipulating the schematics or editing the component parameter data files. A simple command for solution produces a comprehensive transient profile of the parameters of interest. This information may be output to files for hard copy record, or can be displayed on system schematics or plots for presentation or documentation purposes.

The CASE/A program development was funded under NASA Marshall Space Flight Center (MSFC) contract NAS8-36407 and was under continuous development from May 1985 until December 1991. The ARCHIVE enhancement has been under development since February 1992. Sverdrup Technology, Inc., performed the enhancement of porting CASE/A to the SGI from February 1993 through October 1993. The program is based largely on the G189A generalized ECLSS program developed by McDonnell Douglas Astronautics Company under contract to Johnson Space Center (JSC). The architecture of the solution routines has evolved in part from the

G189A and systems improve numerical differencing analyzer (SINDA) programs,[1 2] both of which have been proven in systems level simulations.

The operational characteristics and file structures of the G189A and SINDA models require the user to manage the model data in text format. The SINDA resistor-conductor (R-C) network approach is well suited to applications describing subcomponent level transport phenomena and has been utilized in many systems-level simulations with excellent results. G189A is similar to the large chemical process simulation programs, such as PROCESS and CHEMCAD.[3 4] Though well suited to the systems level simulations targeted for the CASE/A program, they suffer from a rigid data input format and large system overhead. For large models, both SINDA and G189A require considerable management of the input and output data, which experience has shown can consume a large part of the user's time. CASE/A takes the best features of both programs in conjunction with the graphical interface and data management system to produce a user-friendly, interactive tool designed to increase engineering productivity.

The current CASE/A version 5.0 is under NASA/MSFC configuration control, and it is requested that any enhancements or corrections be coordinated with the NASA/MSFC Life Support Branch. The program utilizes very low level graphics operations to allow easy migration to new display systems as described in the CASE/A programmer's manual.[5]

1.2 Program Overview and Capabilities

CASE/A consists of three fundamental parts: (1) the schematic management system, (2) the data base management system, and (3) the simulation control and execution system. In addition, the program provides three other important functional areas: (1) model output management, (2) special system utility commands, and (3) user operations logic capability. An overview of the CASE/A program is shown in figure 1.

The schematic management system allows the user to graphically construct a system model by locating component icons on the screen and connecting the components with physical fluid streams. The graphical interface was developed to take advantage of the gains in productivity that have been obtained through similar CAD/CAE techniques. A complete array of commands are available to allow the user to easily manipulate the screen schematic layout.

The data base management system supports the storage and manipulation of component data, plot utility data, and solution control data. The user can interface with the data bases through interactive edit screens.

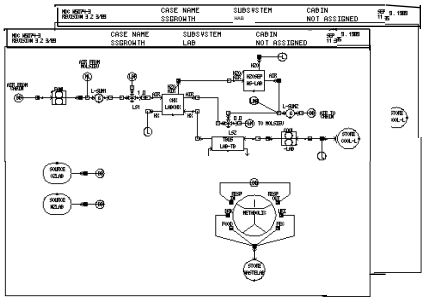
The simulation control and execution system is responsible for initiation and control of the iterative solution process. A time status of the solution progress is displayed on the screen along with any diagnostic messages.

The model output management system provides the user with both tabular and graphical output capability. Complete fluid constituent mass fraction and properties data (mass flow, pressure, temperature, specific heat, density, and viscosity) are generated at user-selected output intervals and are stored in the `CASENAME.LPP` file for user reference. In addition, the CASE/A integrated plot utility (IPU) provides the user with the capability to produce output plots of the above-mentioned fluid constituent data as well as all of the component data base parameters.

Special system utility commands are provided to enable the user to operate more efficiently in the CASE/A environment. For example, commands are provided to change VAX terminal settings, merge CASE/A models across VAX hosts, and assign subsystem heat loads to a particular cabin. An on-line help utility is also incorporated into the program that provides complete user information on CASE/A commands and integrated plot utility commands.

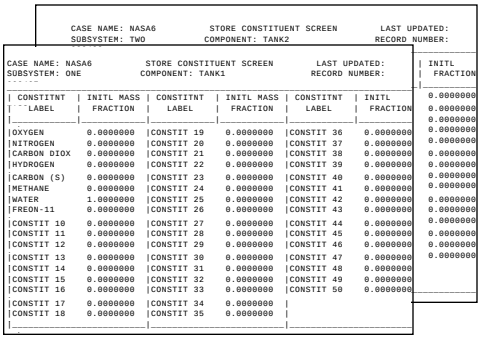
The user is able to customize a CASE/A simulation through optional operations FORTRAN logic. This user-provided code is compiled and linked with a CASE/A model and enables the user to control and timeline component operating parameters during any of seven phases of a model solution: (1) pre-simulation and before component initiations, (2) pre-simulation and after component initiations, (3) before each time step, (4) each solution iteration, (5) after each time step, (6) each output time interval, and (7) postsimulation.

Schematic Management



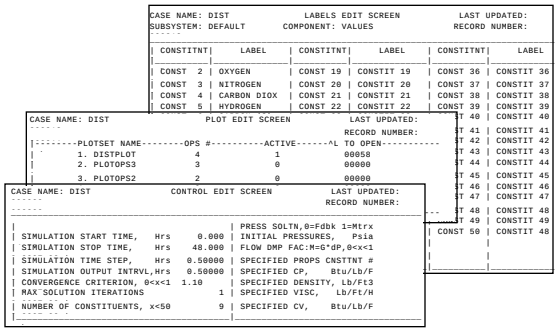
- Locate Components On Subsystem Screens
- Connect Components With Fluid Streams
- Manipulate Screen Layout If Desired

Component Database Management



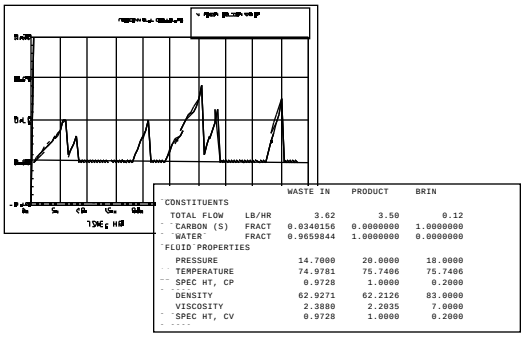
- Component Data Editing And Storage
- Full Screen Or Line Edit Modes Supported

Simulation Control And Execution



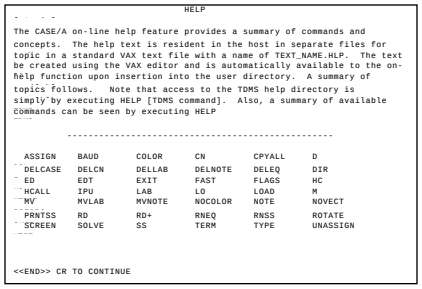
- Simulation Control Constants Setup Screen
- Integrated Plot Utility (IPU) Setup Screen
- Issue SOLVE Command To Initiate Simulation

Model Output Management



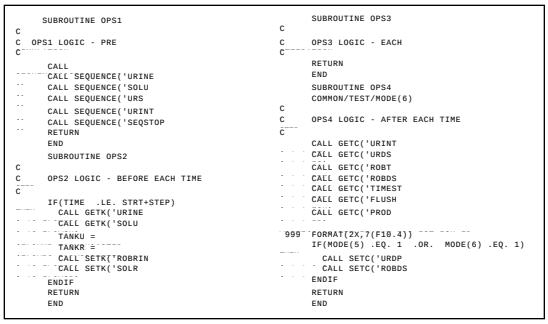
- Transient Data Plots Of User Selected Variables
- Tabular Summaries At Selected Output Intervals

Utility Commands



- On-Line Help Information Screens
- Capability To Merge Models Across VAX Hosts
- Schematic And Component Hardcopy Generation

User Operations Logic



- User-Supplied FORTRAN Logic For Specialized Control
- Component Solution Order Sequencing To Enhance Solution Efficiency

Figure 1. An overview of the CASE/A program.

Each modeled ECLSS/ATCS configuration is referred to as a "case." The creation and modification of a case is managed by the command processor. The user interacts with the command processor to build a schematic of the system or systems under consideration. Each component of the system is represented by a unique icon. Connections between the components define the process flows and interconnections. These connections are input by the user by requesting a connection and then manipulating a crosshair on the screen. The buildup process is shown in figure 2, and consists of placing components on the screen via the locate command (LO) and then connecting (CN) the components as appropriate. All connections between components must be resolved before a solution can be initiated. Commands are provided to adjust the location and configuration of components on the screen to improve the appearance of the layout. Component parameter data are automatically loaded from a default library as each component is located on the screen for a case. The default data are loaded into the component data files and are immediately copied into a new record with the required CASE/A component name and subsystem name. The data can be modified as required through an edit command (E or ED). The default values are based on typical values and give the user a starting point. In most cases, space station development or *Spacelab* hardware has been used as the default data. The system generation and editor commands may be invoked by the user in any meaningful sequence. For example, the user may choose to locate all components first before editing the data bases, or he/she may choose to edit the data base as each component is located.

At the onset of the program construction, it was recognized that some systems may be too complex to display on one screen. The "subsystem screen" concept was developed to allow the model to be partitioned into workable segments for generation and editing. These model segments are invoked with the subsystem command (SS). An interconnect bubble feature is provided to define connections between subsystem screens without the need for a discrete line segment. The program resolves the bubble interconnects by associating components connected to bubbles of the same name. The user may specify as many different bubbles as required to define the system of interest. The bubbles may also be used within a single screen layout to resolve an awkward connection between components.

Another level of hierarchy is available by assigning subsystems to specific cabins or modules with the "ASSIGN" command (see section 6.5). This has the effect of assigning the component heat loads and reference environmental temperatures to a particular cabin/module environment. Up to 25 cabins or modules can be simulated in any given case.

Approximately 400 files are managed by the data base processor for any given configuration in a manner completely transparent to the user. An overview of the CASE/A file structure is provided in figure 3. This file structure was developed from a data base management system developed by MDSSC-Huntsville for use in data reduction and manipulation of engineering data. This system is a relational data base program that utilizes binary data storage. The key elements of the data base system are incorporated into the CASE/A system. The component storage system is described in the CASE/A programmer's manual. The system utilizes centralized file storage, which allows multiple users to use common storage for all cases. If desired, the user can override this option and use private storage. The common storage option has been implemented to allow more efficient utilization of mass storage. The system provides for the required record locking/unlocking and hand shaking protocols to allow simultaneous file access.

Table 1 provides a summary of the CASE/A program capabilities.

The philosophy of the CASE/A program is to provide the user with a collection of basic tools needed to simulate most ECLS and ATCS now under consideration for spacecraft applications. While the program has been designed for flexibility, it is limited to the level of fidelity provided in each component routine. In some cases, the experienced user will require a higher level of fidelity in the operations of a given component. There are two options available to increase the level of fidelity. The first is to utilize the BLACKBOX component to develop a new component. A generic BLACKBOX component is provided to simplify the engineers' task of adding new components, and provides the necessary graphics and data management support.

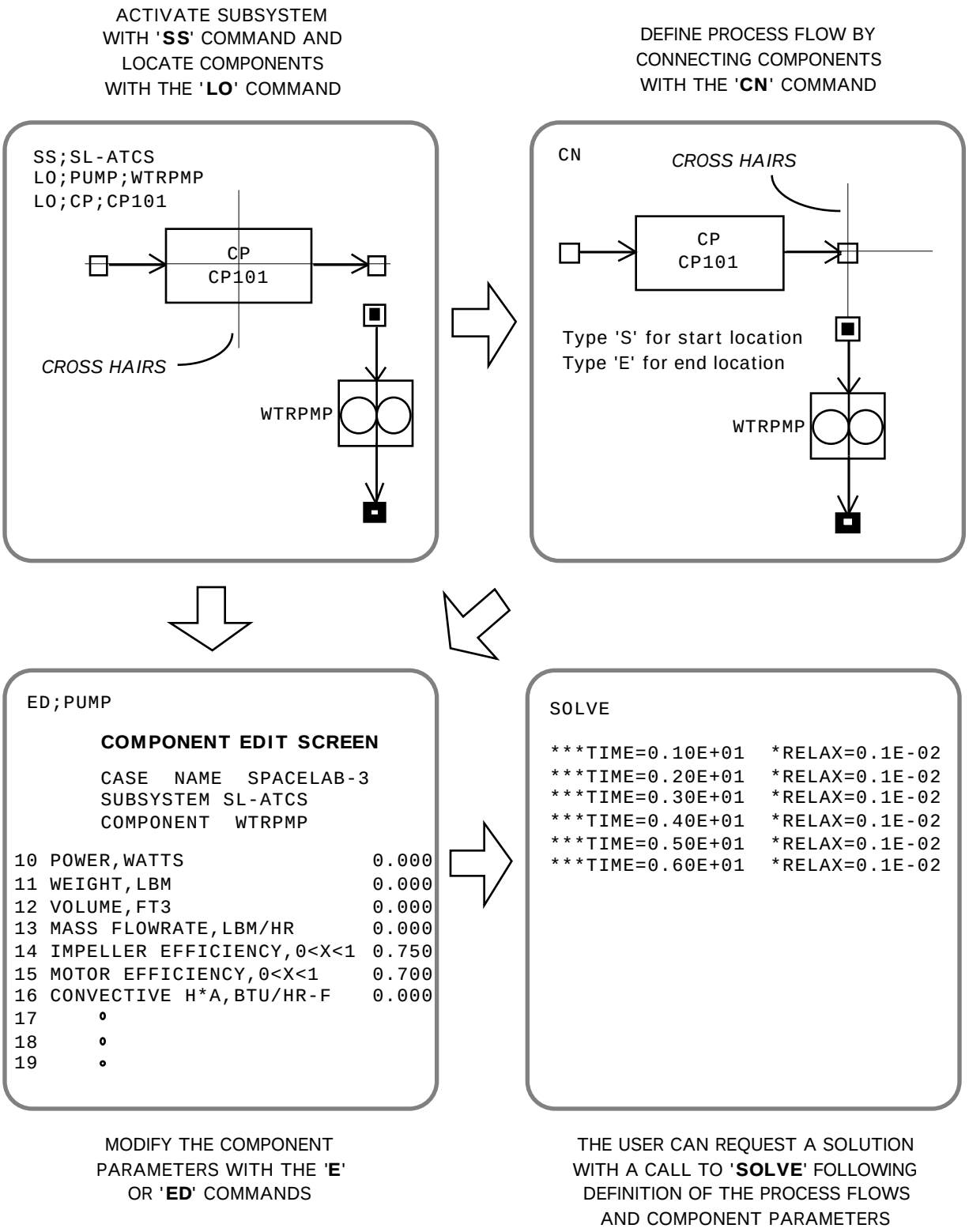


Figure 2. CASE/A model construction process.

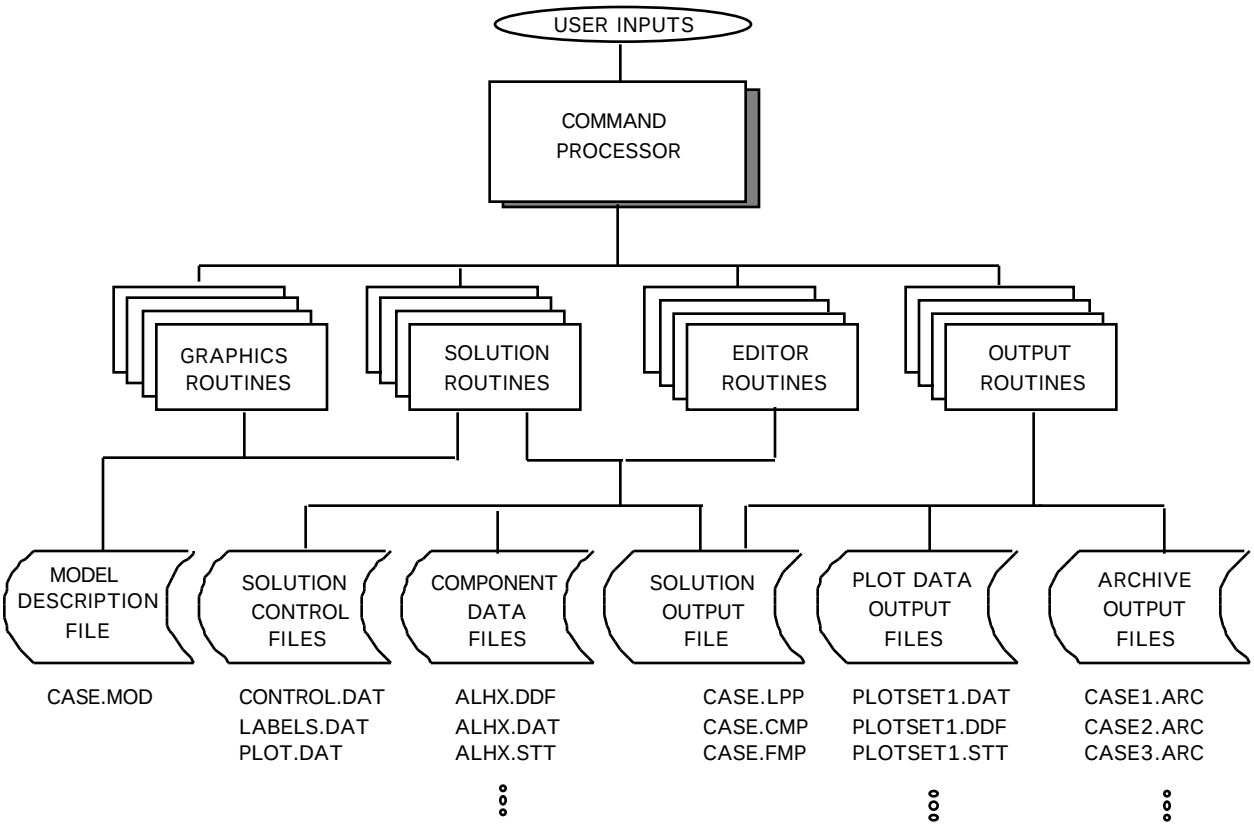


Figure 3. CASE/A file structure.

Table 1. CASE/A program capabilities.

CASE/A Items	Maximum Allowed Per Case
Subsystems	100
Components	1,000
Connections	2,000
Cabins	25
Blackboxes	25
SPLIT components	250
MSPLIT components	250
Constituents	50
Notes	150
Output labels	150

The second option is to modify the existing components to provide the required level of fidelity. In either case, the user must recompile and link the program into a customized executable image file. Command procedures are provided to aid the user in these tasks. The source code is available as a part of this system, and in general, the component logic is relatively well documented. Additional documentation is available in the CASE/A Programmer's Manual.

1.3 Introductory Example Models

This section will introduce the new user to CASE/A with two short introductory sample problems. Only basic CASE/A concepts will be emphasized in these introductory examples. More detailed explanations of commands and functions will be left for description in subsequent sections and in the sample problems found in appendix A.

The first example "Example 1: Case ATC" introduces the new user to the graphic component icon placement and connection using the CASE/A graphic interface.

The second example "Example 2: Case MB6" introduces the new user to the archive retrieve utility, editing the MB6 control file, and running the case solution.

If the user is just getting started using CASE/A, it is recommended to start with Example 1: Case ATC to become familiar with placement and connection of components using the graphic design interface. More advanced users may wish to skip example 1 and start with Example 2: case MB6.

1.3.1 Example 1: Case ATC

The example case "ATC" or active thermal control system will help the reader to understand, from the beginning, how to set up a simple case. The reader will begin by initiating CASE/A and creating a new case and subsystem. Next, component icons are located (positioned) on the screen layout of the subsystem. Finally fluid stream connections are made between these components.

To initiate a new case, ensure that the CASE/A files and your LOGIN.COM file are set up as specified in appendix C (also see section 2.1). Ensure that your current default directory is USERNAME.CASEA (where USERNAME is of course your user ID). Next, link with CASE/A from the user directory to create an executable using the command:

```
$ @CASEA$CODE:LINKCAS
```

Next, invoke CASEA with the command:

```
$RUN/NODEB CASEA
```

At the 'CASEA >' prompt, enter the command NEWCASE to create a new case as follows. The command argument is the name of the example case "ATC":

```
CASEA > NEWCASE ATC
```

(Note: If case "ATC" is already in your CASE/A directory, there will be an error message displayed as follows:

"WARNING: CASE NAME ALREADY EXISTS"

Simply change the name (such as "ATC_1") and retry.

Next, open a subsystem screen, named ATC, with the following command:

```
CASEA_ATC > SS ATC
```

This command creates a blank subsystem screen with the case name, subsystem name, and cabin assignment in the header. *Aside:* Since this subsystem has not been assigned to a cabin, **NONE** should appear in the cabin assignment. Cabin assignments are performed to provide a thermal environment (other than the default environment of 75 °F for structural, convective, and radiant boundary temperatures) and are explained in section 6.5.

Now the user may locate components and connections to create the subsystem ATC as shown in figure 4.

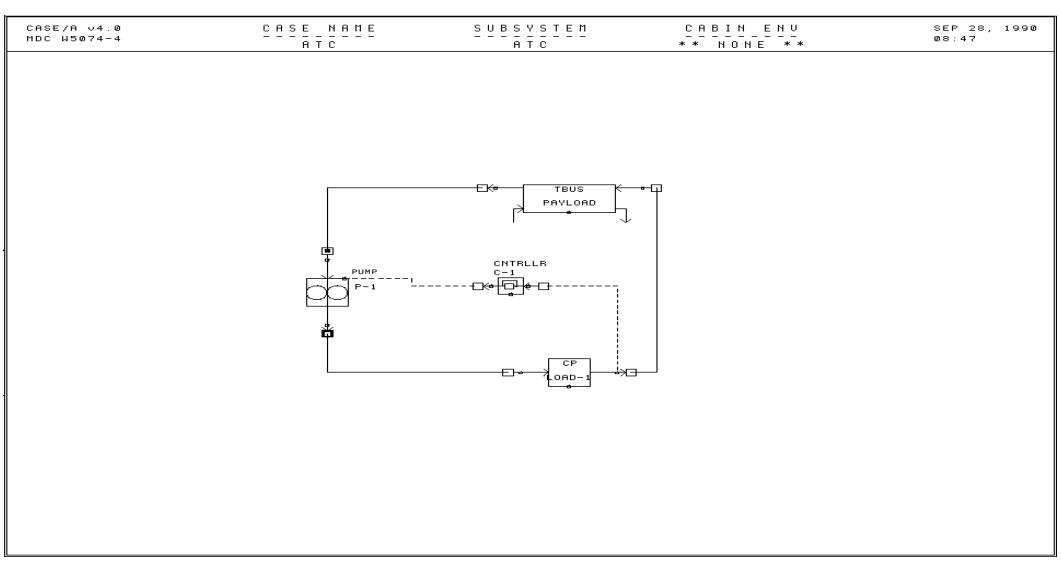


Figure 4. Sample problem 1: ATC subsystem schematic.

The components are located on the subsystem screen with the LO command. First, we will locate the PUMP P-1 on the subsystem screen as follows:

```
CASEA_ATC > LO PUMP P-1
```

The first parameter “PUMP” represents the type of component to be located. The second argument “P-1” is the user-specified component name, and is limited to eight characters and must be unique within this case. After pressing return to enter this command, crosshairs will appear to allow placement of “P-1” on the screen. When the cursor is positioned, hit return or the mouse button to place the component. The graphic icon for “P-1” is shown in figure 5. The small box on each component fluid stream is called a “hit box,” and is used to make fluid stream connections to other components.

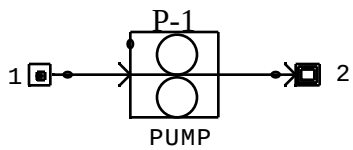


Figure 5. Typical component icon.

To complete the subsystem diagram, locate the other components shown in figure 4. Locate the CP (cold plate) component and name it LOAD-1 as follows:

```
CASEA_ATC > LO CP LOAD-1
```

Locate a TBUS (thermal bus) component and name it PAYLOAD as follows:

```
CASEA_ATC > LO TBUS PAYLOAD
```

Finally, locate a CNTRLLR (controller) component and name it C-1 as follows:

```
CASEA_ATC > LO CNTRLLR C-1
```

Next, the orientation of the components can be adjusted to improve the layout of the schematic. The PUMP component can be rotated using the ROTATE command as follows:

```
CASEA_ATC > ROTATE -90
```

The argument for the ROTATE command must be in multiples of 90˚ and with a magnitude not greater than 360˚. A negative value represents a clockwise rotation. Upon execution of the ROTATE command, the crosshairs will appear. Place the crosshairs on the component to be rotated (the pump) and hit any key or the mouse button. The component will be shown in its new orientation upon execution of the next RD (redraw) command as follows:

```
CASEA_ATC > RD
```

Similarly, rotate the CNTRLLR and TBUS components 180˚ as follows:

```
CASEA_ATC > ROTATE 180
```

Once all of the components have been positioned on the screen, the components are then connected by fluid streams using the CN command. First, we will connect the pump exit to the coldplate inlet with the following command:

```
CASEA_ATC > CN
```

Crosshairs will appear on the terminal screen which can be moved with either the mouse or cursor keys to the “hit box” on the pump exit stream. When positioned over the “hit box” strike any key or the mouse button. This will place the starting point of the stream (note that nothing is shown after this step). To define an intermediate point along the connection stream, position the crosshairs at the selected position and press any key or the mouse button (note that the line segement will now appear). Up to five intermediate points may be defined. For the termination point, the crosshairs are then moved to the “hit box” on the CP inlet stream and any key or mouse button is pressed. This completes the connection process for that stream.

The CN command is also used to connect the coldplate to the thermal bus and the thermal bus to the pump.

The controller “C-1” is connected using the CN command and procedure described above but with the following important differences. As shown in figure 4, the connection path is started at the hit dot on the outlet stream of “LOAD-1” and terminates on the inlet stream of the controller. This is refered to as a “data path” and appears as a dotted line. The controller component is unique in that it accepts data from certain components, acts

upon the data, and may send commands to other components. Please refer to the controller component in section 10 for additional information. Similarly, the output “data path” of “C-1” is connected to the “hit dot” on the pump “P-1.”

At this point, we have introduced the method of model creation in CASE/A. The next example will demonstrate some other CASE/A features. You may complete the Example 1: Case ATC at any time by referring to appendix A.

1.3.2 Example 2: Case MB6

The second example in section 1.3 traces several commands involving case MB6. First, the RETRIEVE process is illustrated, then the DIR command is used to inquire about the components of the case, the SS command is used to view the subsystem schematic of LAYOUT (one of the four subsystems in the case), the CONTROL screen is edited to change the STOP time of the solution to be run, and finally, the SOLVE process of the new model is run.

Before retrieving case MB6, let us examine briefly how this model was first archived. Previously, case MB6 was set up by a user to evaluate carbon dioxide removal in the space station design. The user obtained a solution, and the results and entire model were archived for later retrieval. The information archived includes all input data, OPS code, data resulting from the SOLVE process, and plot data (sections 6.7 and 6.8 describe the archive process in more detail).

With case MB6 loaded, the first step taken to archive the model was to edit the archive data base with the command ED ARCHIVE (this screen is shown in figure 6). Up to 20 archive set names can be listed/entered as shown in figure 6, but during the archive process, only one archive set can be set to active by the switch in the second column. The second screen, the archive data base screen shown in figure 7, can be displayed by entering a ^L (<Ctrl L>) in the rightmost column of the archive edit screen. Here, general information such as the purpose and date of the archived information can be entered and associated with the archived case. Finally, the ARCHIVE command was entered and the user prompted to specify the parts of the model to be archived. The specified files were written to ASCII text files, which can be later retrieved.

Now let us step through the process for retrieving CASE MB6.

First, you should know where the archive directory is located. Two options are described for locating this directory:

(1) The CASE/A convention is to place the archive files in a subdirectory named “ARCH” under the userid.CASEA directory as follows:

```
device:[userid.CASEA.ARCH]
```

If the installation was followed in appendix C, this directory will be created and loaded with the MB6 case. The steps to retrieve the case followed below assume this was done.

```
CASE NAME: MB6                 ARCHIVE EDIT SCREEN          LAST UPDATED: 941217
                                                          RECORD NUMBER: 000003
|-------ARCHIVE SET NAME-------------ACTIVE-------------^L TO OPEN-------------|
|        1. MB6TEST                   1                  00013                |
|        2.                           0                  00000                |
|        3.                           0                  00000                |
|        4.                           0                  00000                |
|        5.                           0                  00000                |
|        6.                           0                  00000                |
|        .                            .                  .                    |
|        .                            .                  .                    |
|        .                            .                  .                    |
|       19.                           0                  00000                |
|       20.                           0                  00000                |
|_____________________________________________________________________________|
```

Figure 6. Archive edit screen for case MB6.

```
CASE NAME: MB6                    ARCHIVE DATA BASE SCREEN                LAST UPDATED: 941217
ARCHIVE SET NAME: MB6TEST                                               RECORD NUMBER: 000013
|-----------------------------------------------------------------------------------------|
|1. TITLE: MB6 TEST                          2 STUDY:  MB6 ARCHIVE TEST FOR VERSION 5.0   |
|                                                                                         |
|               .                 3 MODEL NAME: MB6           4 DURATION:    .1000 HOURS  |
|5 RUN DATE: 12/17/94         8 ENGINEER:   J. KNOX           9 TIME STEP: .016670  HOURS |
|                                                                                         |
|10 PURPOSE: TO TEST ARCHIVE FOR VERSION 5.0.                                             |
|                                                                                         |
|                                                                                         |
|11 MODEL DESCRIPTION:  MB6 IS A COMPLICATED MODEL SIMULATING FOUR SUBSYSTEMS.           .|
|                                                                                         |
|                                                                                         |
|12 RESULTS: THE RESULTS OF THE SOLUTION COMPARE FAVORABLY.                     .         |
|                                                                                         |
|                                                                                         |
|13 RUN SPECIFIC   THE STOP TIME IS SHORTENED TO .1 HOURS (6 MINUTES) FROM THE FULL.RUN   |
| MODEL PARAMETERS:TO SHOW AN EXAMPLE RUN.                                                |
|                                                                                         |
|                                                                                         |
|                                                                                         |
|_________________________________________________________________________________________|
```

Figure 7. Archive data base screen for case MB6 and archive set MB6TEST.

(2) On the other hand, the archive utility also provides the option to specify another archive directory from which to retrieve an archived model. Using this method, case MB6 can be retrieved from the following file:

device:[CASEA.USER.ARCH]
where **device** is the device name for the location of the CASE/A directories.

Or the user may wish to specify other archive directories (such as another user's directory or common user archive) to retrieve case MB6. The system manager should be consulted if you have trouble locating the correct archive files.

After locating the archive directory, a new case must be created containing all data from MB6. We will call it "MB6_1." During the retrieve process which follows, the information in the ASCII archive file is copied into this new case. Prompts request the user to specify what information should be included in the new model as shown in figure 8.

To start, initiate CASE/A, then enter the following command to create the new case:

```
CASEA > NEWCASE MB6_1
```

Next enter the retrieve command as follows:

```
CASEA > RETRIEVE
```

Figure 8 shows the dialog after entering the RETRIEVE command. In the figure, the user's responses are typed in bold print. Use one of the two options described above for specifying the archive directory. The archive files are identified with the "ARC" suffix.

Enter the "source case name": "MB6" – the first part of the file name.

Enter the "archive set name": "EXAMPLE2" – the second part of the file name.

Merge the archive case with the new case by selecting "1" – Subsystem and Control, Labels, Plot, Usercon, and Archive and follow figure 8 to complete the retrieval.

After completing the retrieve, exit CASE/A, then initiate the new case MB6_1 by linking with CASE/A from the user's directory (i.e., your CASE/A directory). Note that the OPS code for MB6_1 (MB6_1.FOR) is now located in the

```
CASEA_MB6_X > RETRIEVE

---------------------RETRIEVE UTILITY--------------------
This utility provides the capability to retrieve a model,
IPU plot data, and associated ops code into the currently
specified case.  Retrieval of this file can only be made
from a valid archive file (please refer to users manual).

Press <ret> to continue or -1 to abort.
<return>
The default archive subdirectory is[QUINNHE.CASEA.MB6.ARCH]

DO YOU WISH TO USE THE DEFAULT ARCHIVE DIRECTORY(Y or N)
Enter Y for yes or N for No:
Y


A LISTING OF THE ARCHIVED FILES FOLLOWS:

      FILE             DATE CREATED

MB6_TST_MB6_HAL.ARC;1
                      7-SEP-1995 16:05:37.39
MB6_X_EXAMPLE2.ARC;1
                     11-JAN-1996 15:12:38.34

Total of 2 files.
%DELETE-I-FILDEL, USER3$:[QUINNHE.CASEA.MB6.ARCH]ARC.LST;1 deleted (4 blocks)


ENTER THE SOURCE CASE NAME (e.g. MOD_1 of MOD_1_SET1.ARC):
MB6_X

ENTER THE ARCHIVE SET NAME (e.g. SET1 of MOD_1_SET1.ARC):
EXAMPLE2

 A CASE CAN BE MERGED IN SEVERAL PARTS.  PLEASE SELECT ONE:

 1)  SUBSYSTEM(S) AND CONTROL, LABELS, PLOT, USERCON AND ARCHIVE
 2)  SUBSYSTEM(S) ONLY
 3)  CONTROL, LABELS, PLOT, USERCON AND/OR ARCHIVE ONLY
 OR -1 TO ABORT...

PLEASE CHOOSE ONE: 1

A LISTING OF THE SOURCE CASE SUBSYSTEMS FOLLOWS:

 LABA
 ORBITER
 NODE2
 LAYOUT




ENTER SS NAME, <RET> FOR ALL, OR -1 TO ABORT:
<return>
MERGING SOURCE CASE EQUIP DATA INTO LOCAL CASE...
DATA CREATED FOR: LAWASTE   TYPE: STORE     SS: LABA
DATA CREATED FOR: ORBWASTE  TYPE: STORE     SS: ORBITER
DATA CREATED FOR: N2WASTE   TYPE: STORE     SS: NODE2
DATA CREATED FOR: O2ORB     TYPE: STORE     SS: ORBITER
DATA CREATED FOR: N2ORB     TYPE: STORE     SS: ORBITER
DATA CREATED FOR: ORBWAST   TYPE: STORE     SS: ORBITER
DATA CREATED FOR: LABACHX   TYPE: CHX       SS: LABA
DATA CREATED FOR: ORBCHX    TYPE: CHX       SS: ORBITER
DATA CREATED FOR: N2CHX     TYPE: CHX       SS: NODE2
DATA CREATED FOR: WS-LABA   TYPE: H2OSEP    SS: LABA
DATA CREATED FOR: WS-ORB    TYPE: H2OSEP    SS: ORBITER
DATA CREATED FOR: WS-N2     TYPE: H2OSEP    SS: NODE2
DATA CREATED FOR: ORBO2N2   TYPE: O2N2      SS: ORBITER
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: LABA
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: LAYOUT
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: LABA
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: LABA
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: LAYOUT
DATA CREATED FOR: N2        TYPE: BUBBLE    SS: LAYOUT
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: N2        TYPE: BUBBLE    SS: NODE2
DATA CREATED FOR: N2        TYPE: BUBBLE    SS: NODE2
DATA CREATED FOR: N2        TYPE: BUBBLE    SS: NODE2
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: LABA
DATA CREATED FOR: LA        TYPE: BUBBLE    SS: LABA
```

Figure 8. Introductory example retrieve process.

```
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: N2        TYPE: BUBBLE    SS: LAYOUT
DATA CREATED FOR: N2        TYPE: BUBBLE    SS: LAYOUT
DATA CREATED FOR: TB        TYPE: BUBBLE    SS: NODE2
DATA CREATED FOR: TB        TYPE: BUBBLE    SS: LABA
DATA CREATED FOR: TB        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: OR        TYPE: BUBBLE    SS: ORBITER
DATA CREATED FOR: ORLIOH1   TYPE: LIOH      SS: ORBITER
DATA CREATED FOR: ORLIOH2   TYPE: LIOH      SS: ORBITER
DATA CREATED FOR: LAS1      TYPE: SPLIT     SS: LABA
DATA CREATED FOR: ORBS1     TYPE: SPLIT     SS: ORBITER
DATA CREATED FOR: N2S1      TYPE: SPLIT     SS: NODE2
DATA CREATED FOR: ORBS3     TYPE: SPLIT     SS: ORBITER
DATA CREATED FOR: LAS2      TYPE: SPLIT     SS: LABA
DATA CREATED FOR: LA-CREW1  TYPE: CREW      SS: LABA
DATA CREATED FOR: ORBCREW1  TYPE: CREW      SS: ORBITER
DATA CREATED FOR: N2CREW1   TYPE: CREW      SS: NODE2
DATA CREATED FOR: ORBCREW2  TYPE: CREW      SS: ORBITER
DATA CREATED FOR: ORBITER   TYPE: CABIN     SS: LAYOUT
DATA CREATED FOR: PBM1      TYPE: CABIN     SS: LAYOUT
DATA CREATED FOR: IMV-N2LA  TYPE: PUMP      SS: LAYOUT
DATA CREATED FOR: THC-LA    TYPE: PUMP      SS: LABA
DATA CREATED FOR: THC-ORB   TYPE: PUMP      SS: ORBITER
DATA CREATED FOR: THC-N2    TYPE: PUMP      SS: NODE2
DATA CREATED FOR: IMV-LAN2  TYPE: PUMP      SS: LAYOUT
DATA CREATED FOR: IMV-N2PB  TYPE: PUMP      SS: LAYOUT
DATA CREATED FOR: IMV-PBN2  TYPE: PUMP      SS: LAYOUT
DATA CREATED FOR: LA-SUM2   TYPE: SUM       SS: LABA
DATA CREATED FOR: ORBSUM2   TYPE: SUM       SS: ORBITER
DATA CREATED FOR: N2-SUM2   TYPE: SUM       SS: NODE2
DATA CREATED FOR: ORBARSUM  TYPE: SUM       SS: ORBITER
DATA CREATED FOR: LA-SUM1   TYPE: SUM       SS: LABA
DATA CREATED FOR: 4BMSIN    TYPE: SUM       SS: LABA
DATA CREATED FOR: N2_SUM1   TYPE: SUM       SS: NODE2
DATA CREATED FOR: TBSUM     TYPE: SUM       SS: NODE2
DATA CREATED FOR: ORBARSP   TYPE: MSPLIT    SS: ORBITER
DATA CREATED FOR: 4BMS      TYPE: BLACKBOX  SS: LABA
DATA CREATED FOR: RCRS      TYPE: BLACKBOX  SS: ORBITER
DATA CREATED FOR: H1        TYPE: HATCH     SS: LAYOUT
DATA CREATED FOR: H2        TYPE: HATCH     SS: LAYOUT
DATA CREATED FOR: H3        TYPE: HATCH     SS: LAYOUT
DATA CREATED FOR: O2LAB     TYPE: SOURCE    SS: LABA
DATA CREATED FOR: N2LAB     TYPE: SOURCE    SS: LABA
DATA CREATED FOR: N2COOL    TYPE: SOURCE    SS: NODE2
DATA CREATED FOR: LACOOL    TYPE: SOURCE    SS: LABA
DATA CREATED FOR: ORBCOOL   TYPE: SOURCE    SS: ORBITER
DATA CREATED FOR: SPACE-L   TYPE: SINK      SS: LABA
DATA CREATED FOR: ORBCOND   TYPE: SINK      SS: ORBITER
DATA CREATED FOR: LABACOND  TYPE: SINK      SS: LABA
DATA CREATED FOR: N2COND    TYPE: SINK      SS: NODE2
DATA CREATED FOR: TBS       TYPE: SINK      SS: NODE2
DATA CREATED FOR: VENT      TYPE: SINK      SS: ORBITER
DATA CREATED FOR: LABA      TYPE: MODULE    SS: LAYOUT
DATA CREATED FOR: NODE2     TYPE: MODULE    SS: LAYOUT

 DO YOU WANT TO IMPORT THE LABELS RECORD (Y/N)?
(THIS WILL OVERWRITE EXISTING RECORD): Y
LABELS ASCII IMPORT COMPLETE.

 DO YOU WANT TO IMPORT THE CONTROL RECORD (Y/N)?
(THIS WILL OVERWRITE EXISTING RECORD): Y
CONTROL ASCII IMPORT COMPLETE.

 DO YOU WANT TO IMPORT THE USERCON RECORD (Y/N)?
(THIS WILL OVERWRITE EXISTING RECORD): Y
USERCON ASCII IMPORT COMPLETE.

 DO YOU WANT TO IMPORT THE PLOT RECORD (Y/N)?
(THIS WILL OVERWRITE EXISTING RECORD): Y
PLOT ASCII IMPORT COMPLETE.

 DO YOU WANT TO IMPORT THE ARCHIVE RECORD (Y/N)?
(THIS WILL OVERWRITE EXISTING RECORD): Y
ARCHIVE ASCII IMPORT COMPLETE.

ASCII MERGE COMPLETED: NO. OF IMPORTED COMPONENTS =           83
CHECKING FOR ACTIVE PLOTSETS...

DO YOU WANT TO RETRIEVE THE DATA ASSOCIATED WITH THE SELECTED PLOTSETS(Y/N)?
(THIS WILL CREATE TWO NEW FILES FOR EACH PLOTSET): Y

COPYING ARCHIVED IPU PLOT DATA TO FILE...
%COPY-S-COPIED, USER3$:[QUINNHE.CASEA.MB6.ARCH]MB6_X_EXAMPLE2_CABPLOT.DAT;1 cop)
%COPY-S-COPIED, USER3$:[QUINNHE.CASEA.MB6.ARCH]MB6_X_EXAMPLE2_CABPLOT.STT;1 cop)

DO YOU WANT TO RETRIEVE THE OPS CODE (Y/N)?
( A NEW FILE WILL BE CREATED FOR THIS CASE): Y
OPS CODE MERGE COMPLETE.
CASEA_MB6_X >
```

Figure 8. Introductory example retrieve process (continued).

user's directory and must first be compiled with the command:

```
$ FOR/NOOPT MB6_1
```

Next, use the following link command to create the executable (Note: this assumes that the code was installed according to the instructions in appendix C):

```
$ @CASE$CODE:LINKCAS MB6_1
```

After linking with CASE/A code to create an executable in the user directory, the program is invoked with the "RUN" command:

```
$ RUN/NODEB MB6_1
```

Next, load the case MB6_1 and use the DIR command to show the subsystems of the model. As shown in figure 9, the system MB6_1 consists of four subsystems: LAYOUT, LABA, ORBITER, and NODE2.

The user may wish to view a subsystem schematic of case MB6_1. The subsystem to be analyzed is shown in CASE/A schematic form in figure 10. The subsystem LAYOUT consists of 16 components: two cabins, two modules, four pumps, and three hatches connecting the cabins. (Note that a cabin has a single air volume in contrast to a module which has three air volumes.) There are also five bubbles that facilitate the interconnection of flow streams on different subsystem screens. The LAYOUT subsystem is a pumped air system with hatch components for pressure relief between the cabins and the modules of the space station. The other subsystems, LAB, ORBITER, and NODE2 (these can be viewed by entering the SS command) contain the temperature and humidity control and CO_2 removal equipment for those cabins. This model has been used in evaluating the carbon dioxide removal assembly (CDRA) requirements for the space station during a mission build flight. (For more detailed information on this model see reference 6).

When all of the connections are verified, the user may edit the component data bases if desired. If the default values (see section 10, Component Descriptions) for a particular component are acceptable to the user, then there is no need to edit the data base for that component. The model will run correctly with the components as currently set. It is recommended to the run the model first with default parameters, then subsequent runs can be more experimental with changes to the component data bases.

If the user desires to update the components, the data bases are accessed through the E or ED commands. The E command allows the user to edit a component data base by placing the crosshairs on the component icon and then typing any character on the keyboard. The user may also access the component data without the subsystem schematic screen by using the ED command as follows:

```
TERM 2    [sets terminal to VT-100 mode]
ED PUMP
```

Since several pumps exist in MB6_1, a menu of pumps is displayed that allows selection of the specific pump to edit (fig. 11). Upon selection, the pump edit screen appears, as shown in figure 11, with default data. The pump flowrate and other input parameters may be changed by tabbing the cursor through the edit screen and entering the desired numerical data. To exit the edit screen, type <Ctrl Z> to save the data or ESC (<Ctrl X>) not to save the data. The edit screens are accessed in a similar manner for the rest of the components in the case. Further explanations are provided in sections 3 and 10.

Once satisfied with the component data base settings, edit the CASE/A system control parameters, shown in figure 12, as follows:

```
ED CONTROL
```

Start time, stop time, time step, output interval, convergence criteria, etc., are entered before initializing a simulation solution. It is recommended to alter the stop time in the control file to shorten the solution run time for this exercise. Change the stop time to 1.0 h in preparation for the next step. Type ^Z (control-Z) to save the change and exit.

```
CASEA_MB6_1 > DIR
_________________________________________________________________________________
|                              SUBSYSTEM LISTING                                  |
|   SUBSYS  CABIN       SUBSYS  CABIN       SUBSYS  CABIN      SUBSYS CABIN      |
|   ------  -----       ------  -----       ------  -----      ------  -----     |
|   LAYOUT  ........     LABA  LABA       ORBITER  ORBITER     NODE2  NODE2      |
|                                                                                 |
|_________________________________________________________________________________|

<RET> = NEXT, SUBSYS NAME = SHOW LISTING, 0 = MAIN MENU, -1 = QUIT:


_________________________________________________________________________________
|         SUBSYSTEM: LAYOUT                CABIN ENVIRONMENT: ........            |
|                                                                                 |
|     NAME  TYPE          NAME  TYPE          NAME  TYPE         NAME  TYPE       |
|     ----  ----          ----  ----          ----  ----         ----  ----       |
|  ORBITER  CABIN         PBM1  CABIN         LABA  MODULE      NODE2  MODULE     |
| IMV-N2LA  PUMP      IMV-LAN2  PUMP      IMV-N2PB  PUMP     IMV-PBN2  PUMP      |
|       H1  HATCH           H2  HATCH           H3  HATCH          LA  BUBBLE     |
|       OR  BUBBLE          N2  BUBBLE          N2  BUBBLE         N2  BUBBLE     |
|_________________________________________________________________________________|

<RET> = NEXT, SUBSYS NAME = SHOW LISTING, 0 = MAIN MENU, -1 = QUIT:
_


_________________________________________________________________________________
|         SUBSYSTEM: LABA                  CABIN ENVIRONMENT: LABA                |
|                                                                                 |
|     NAME  TYPE          NAME  TYPE          NAME  TYPE         NAME  TYPE       |
|     ----  ----          ----  ----          ----  ----         ----  ----       |
|  LAWASTE  STORE        O2LAB  SOURCE       N2LAB  SOURCE     LACOOL  SOURCE     |
|   THC-LA  PUMP      LA-CREW1  CREW       LABACHX  CHX       WS-LABA  H2OSEP     |
|     LAS1  SPLIT         LAS2  SPLIT      LA-SUM2  SUM       LA-SUM1  SUM        |
|   4BMSIN  SUM           4BMS  BLACKBOX   SPACE-L  SINK     LABACOND  SINK       |
|       LA  BUBBLE          LA  BUBBLE          LA  BUBBLE         LA  BUBBLE     |
|       LA  BUBBLE          TB  BUBBLE                                             |
|_________________________________________________________________________________|

<RET> = NEXT, SUBSYS NAME = SHOW LISTING, 0 = MAIN MENU, -1 = QUIT:

_


_________________________________________________________________________________
|         SUBSYSTEM: ORBITER               CABIN ENVIRONMENT: ORBITER             |
|                                                                                 |
|     NAME  TYPE          NAME  TYPE          NAME  TYPE         NAME  TYPE       |
|     ----  ----          ----  ----          ----  ----         ----  ----       |
| ORBWASTE  STORE        O2ORB  STORE        N2ORB  STORE     ORBWAST  STORE      |
|  ORBCOOL  SOURCE     THC-ORB  PUMP      ORBCREW1  CREW     ORBCREW2  CREW       |
|  ORBO2N2  O2N2        ORBCHX  CHX         WS-ORB  H2OSEP    ORLIOH1  LIOH       |
|  ORLIOH2  LIOH         ORBS1  SPLIT        ORBS3  SPLIT     ORBSUM2  SUM        |
| ORBARSUM  SUM        ORBARSP  MSPLIT        RCRS  BLACKBOX  ORBCOND  SINK       |
|     VENT  SINK            OR  BUBBLE          OR  BUBBLE         OR  BUBBLE     |
|       LA  BUBBLE          OR  BUBBLE          TB  BUBBLE         OR  BUBBLE     |
|       OR  BUBBLE                                                                |
|_________________________________________________________________________________|

<RET> = NEXT, SUBSYS NAME = SHOW LISTING, 0 = MAIN MENU, -1 = QUIT:


_________________________________________________________________________________
|         SUBSYSTEM: NODE2                 CABIN ENVIRONMENT: NODE2               |
|                                                                                 |
|     NAME  TYPE          NAME  TYPE          NAME  TYPE         NAME  TYPE       |
|     ----  ----          ----  ----          ----  ----         ----  ----       |
|  N2WASTE  STORE       N2COOL  SOURCE      THC-N2  PUMP      N2CREW1  CREW       |
|    N2CHX  CHX          WS-N2  H2OSEP        N2S1  SPLIT     N2-SUM2  SUM        |
|  N2_SUM1  SUM          TBSUM  SUM         N2COND  SINK          TBS  SINK       |
|       N2  BUBBLE          N2  BUBBLE          N2  BUBBLE         TB  BUBBLE     |
|_________________________________________________________________________________|

DIRECTORY LISTING COMPLETE.
CASEA_MB6 >
```

Figure 9. Example DIR command response.

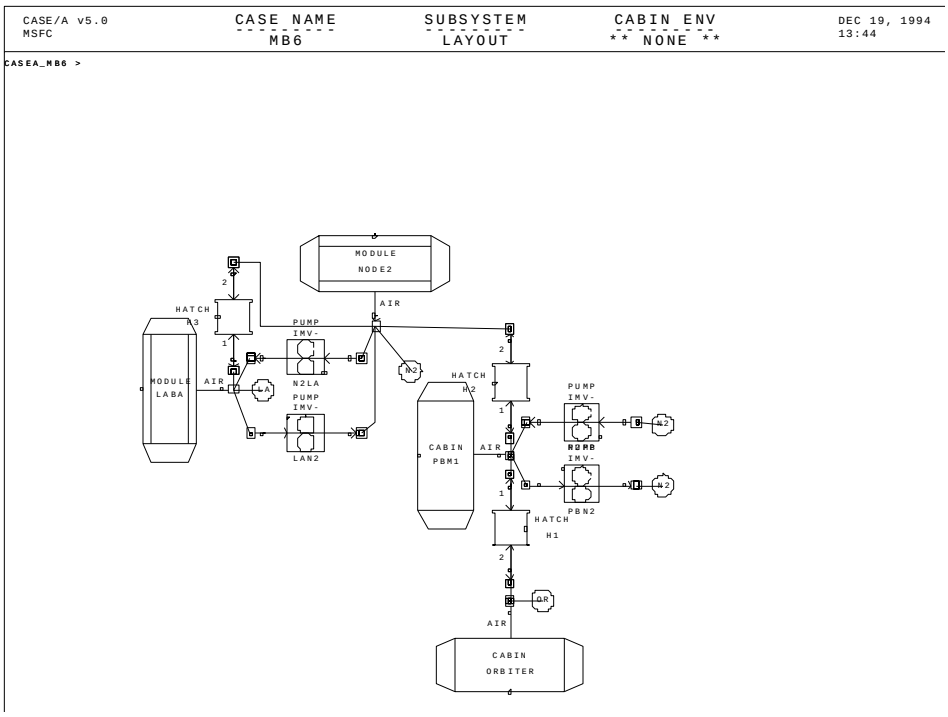


Figure 10. Example model schematic of subsystem LAYOUT.

COMPONENT PARAMETERS EDIT SCREEN MENU

NO.	SUBSYSTEM NAME	COMPONENT NAME
1	LAYOUT	IMV-N2LA
2	LABA	THC-LA
3	ORBITER	THC-ORB
4	NODE2	THC-N2
5	LAYOUT	IMV-LAN2
6	LAYOUT	IMV-N2PB
7	LAYOUT	IMV-PBN2

PLEASE SELECT FROM THE EQUIP LIST ABOVE
BY DESIGNATED NUMBER OR ZERO TO ABORT

CASE NAME: MB6 PUMP EDIT SCREEN LAST UPDATED: 920602
SUBSYSTEM: LAYOUT COMPONENT: IMV-N2LA RECORD NUMBER: 001301

** GENERAL INPUT DATA **		** THERMAL CHARACTERISTICS DATA **		** CENTRIFUGAL PUMP OPTION CONSTANTS **	
10 MASS FLOWRATE (LBM/HR)	0.000	22 SHEL-ENV CONVCT (BTU/H/F)	0.200	35 FLOW DAMPING FACT (0<X<1)	0.10000000
11 INLET PRESSURE (PSIA)	0.000	23 SHEL-ENV RADIAT FAE (FT2)	0.800	13 CHARCTRISTIC CURVE CON A0	0.00000E+00
12 OUTLET PRESSURE (PSIA)	0.000	24 SHEL-ENV CONDCT (BTU/H/F)	1.000	14 CHARCTRISTIC CURVE CON A1	0.00000E+00
18 IMPELLER EFFICIENCY (0<X<1)	1.000	25 MASS-SHEL COND (BTU/H/F)	1.000	15 CHARCTRISTIC CURVE CON A2	0.00000E+00
19 MOTOR EFFICIENCY (0<X<1)	1.000	26 WALL-MASS COND (BTU/H/F)	50.000	16 CHARCTRISTIC CURVE CON A3	0.00000E+00
20 SPECIFIED MOTR QFLAG (0=N,1=Y)	0.000	27 FLUID-WALL EFFECT (0<X<1)	0.899	17 CHARCTRISTIC CURVE CON A4	0.00000E+00
21 SPECIFIED MOTOR QLOAD (BTU/HR)	150.000	28 THERMAL CAPACTNCE (BTU/F)	0.000	36 ** OPEN INPUT LOCATION **	0.000
34 FLUID CODE(0=?, 2-50=CNSTIT #)	0.000	29 INITIAL MASS TEMP (F)	75.000	7 POWER (WATTS)	1.142
32 ** OPEN INPUT LOCATION **	0.000	30 THERMAL RELAX CRITERIA	0.0000100	8 WEIGHT (LBM)	0.000
33 ** OPEN INPUT LOCATION **	0.000	31 THERMAL SOLUTN MAX COUNT	100.000	9 VOLUME (FT^3)	0.000

** BENCHMARK DATA **	MIN	NOM	MAX
PUMP INLET PRESS (PSIA)	10.1252	10.1646	10.2856
PUMP OUTLET PRESS (PSIA)	10.1271	10.1631	10.3117
PUMP PRESS RISE (PSIA)	3.8147E-6	.0041954	0.0430
PUMP FLOW RATE (LBM/HR)	385.2323	386.7937	396.7657
PUMP POWER (WATTS)	0.0000	0.5715	17.4609

** OUTPUT DATA **		
37 MAX CAPACITY FLOW RATE	(LBM/HR)	0.000
38 PRESENT PRESSURE RISE	(PSID)	0.000
39 PRESENT FLOW CONVERGENCE FLAG		0.000
40 PRESENT MASS TEMPERATURE	(F)	84.627
41 PRESENT SHELL TEMPERATURE	(F)	78.161
42 ** OPEN OUTPUT LOCATION **		0.000

Figure 11. Example component edit screen—menu.

```
CASE NAME: MB6                CONTROL EDIT SCREEN          LAST UPDATED: 931216
                                                         RECORD NUMBER: 000003
_____________________________________________________________________________
| TIME UNIT FLAG (1=Hr,2=Min,3=Sec)    1 | PRESS SOLTN,0=Fdbk 1=Mtrx        1. |
| START TIME (Hr,Min, or Sec)      0.000 | INITIAL PRESSURES,   Psia   10.200 |
| STOP TIME (Hr, Min, or Sec)      1.000 | FLOW DMP FAC:M=G*dP,0<x<1 0.000E+00 |
| TIME STEP (Hr, Min, or Sec)    0.01667 | SPECIFIED PROPS CNSTTNT #        0. |
| OUTPUT INTRVAL(H,M, or S)     12.00000 | SPECIFIED CP,     Btu/Lb/F 0.000E+00 |
| PLOT TIME UNITS (1=Hr,2=Min,3+Sec)   1 | SPECIFIED DENSITY, Lb/Ft3 0.000E+00 |
| CONVERGENCE CRITERION, 0<x<1 0.500E-03 | SPECIFIED VISC,    Lb/Ft/H 0.000E+00 |
| MAX SOLUTION ITERATIONS             50 | SPECIFIED CV,     Btu/Lb/F 0.000E+00 |
| NUMBER OF CONSTITUENTS, x<50         9 |                                    |
|_______________________________________|____________________________________|
```

Figure 12. Example CONTROL edit screen.

The SOLVE command may now be issued to initiate the solution routine. The following messages will be sent to the screen to inform the user of the simulation status:

```
CASEA_MB6_1 > solve
READING SIMULATION CONTROL DATA...
CHECKING FOR ACTIVE PLOTSETS...
READING COMPONENT DATA BASE PARAMETERS...
CONSTRUCTING PSEUDO-COMPUTE SEQUENCE & FLOW MAP...

 NOTE:
  THE FOLLOWING SUBSYSTEMS HAVE NOT BEEN ASSIGNED
  TO A CABIN FOR ENVIRONMENTAL HEAT EXCHANGE:

* LAYOUT
*
*
*
*

   THE EXTERNAL HEAT EXCHANGE OF THE COMPONENTS IN
   THESE SUBSYSTEMS WILL BE IGNORED IN THE SOLUTION.
   CONTINUING SIMULATION...


PERFORMING COMPONENT INITIALIZATION OPERATIONS...
INITIATING SIMULATION SOLUTION...
**** TIME= 0.0000E+00 HRS  RELX= 0.5000E-03  RXCC= 0.0000E+00  LOOPS=   0
**** TIME= 0.1667E-01 HRS  RELX= 0.5000E-03  RXCC= 0.2008E-03  LOOPS=  39
**** TIME= 0.3334E-01 HRS  RELX= 0.5000E-03  RXCC= 0.1061E-03  LOOPS=  27
**** TIME= 0.5001E-01 HRS  RELX= 0.5000E-03  RXCC= 0.9740E-04  LOOPS=  12
**** TIME= 0.6668E-01 HRS  RELX= 0.5000E-03  RXCC= 0.2311E-03  LOOPS=   8
**** TIME= 0.8335E-01 HRS  RELX= 0.5000E-03  RXCC= 0.3114E-03  LOOPS=  16
**** TIME= 0.1000E+00 HRS  RELX= 0.5000E-03  RXCC= 0.4314E-03  LOOPS=   4
**** TIME= 0.1167E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2140E-03  LOOPS=   6
**** TIME= 0.1334E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3423E-03  LOOPS=   5
**** TIME= 0.1500E+00 HRS  RELX= 0.5000E-03  RXCC= 0.4267E-03  LOOPS=   5
**** TIME= 0.1667E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2278E-03  LOOPS=   6
**** TIME= 0.1834E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2174E-03  LOOPS=   6
**** TIME= 0.2000E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3864E-03  LOOPS=   6
**** TIME= 0.2167E+00 HRS  RELX= 0.5000E-03  RXCC= 0.1967E-03  LOOPS=   6
**** TIME= 0.2334E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2608E-03  LOOPS=   5
**** TIME= 0.2501E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2372E-03  LOOPS=   5
**** TIME= 0.2667E+00 HRS  RELX= 0.5000E-03  RXCC= 0.6460E-04  LOOPS=  13
**** TIME= 0.2834E+00 HRS  RELX= 0.5000E-03  RXCC= 0.4636E-03  LOOPS=   4
**** TIME= 0.3001E+00 HRS  RELX= 0.5000E-03  RXCC= 0.4171E-03  LOOPS=   4
**** TIME= 0.3167E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3800E-03  LOOPS=   4
**** TIME= 0.3334E+00 HRS  RELX= 0.5000E-03  RXCC= 0.4734E-03  LOOPS=   8
**** TIME= 0.3501E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3042E-03  LOOPS=   4
**** TIME= 0.3667E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2900E-03  LOOPS=   3
**** TIME= 0.3834E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3534E-03  LOOPS=   3
**** TIME= 0.4001E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3280E-03  LOOPS=   4
**** TIME= 0.4167E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2887E-03  LOOPS=   3
**** TIME= 0.4334E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2860E-03  LOOPS=   3
**** TIME= 0.4501E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2700E-03  LOOPS=   4
**** TIME= 0.4668E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3045E-03  LOOPS=   3
**** TIME= 0.4834E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3024E-03  LOOPS=   3
**** TIME= 0.5001E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3017E-03  LOOPS=   3
**** TIME= 0.5168E+00 HRS  RELX= 0.5000E-03  RXCC= 0.1543E-03  LOOPS=   4
**** TIME= 0.5334E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2969E-03  LOOPS=   3
**** TIME= 0.5501E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2933E-03  LOOPS=   3
**** TIME= 0.5668E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2900E-03  LOOPS=   3
**** TIME= 0.5835E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2880E-03  LOOPS=   3
**** TIME= 0.6001E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2864E-03  LOOPS=   3
**** TIME= 0.6168E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2845E-03  LOOPS=   3
**** TIME= 0.6335E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2823E-03  LOOPS=   3
**** TIME= 0.6501E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3080E-03  LOOPS=   3
**** TIME= 0.6668E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2780E-03  LOOPS=   3
**** TIME= 0.6835E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2768E-03  LOOPS=   3
**** TIME= 0.7001E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2756E-03  LOOPS=   3
**** TIME= 0.7168E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2781E-03  LOOPS=   3
**** TIME= 0.7335E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2851E-03  LOOPS=   3
**** TIME= 0.7501E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2895E-03  LOOPS=   3
**** TIME= 0.7668E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2947E-03  LOOPS=   3
**** TIME= 0.7835E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2997E-03  LOOPS=   3
**** TIME= 0.8002E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3060E-03  LOOPS=   3
**** TIME= 0.8168E+00 HRS  RELX= 0.5000E-03  RXCC= 0.3030E-03  LOOPS=   3
**** TIME= 0.8335E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2604E-03  LOOPS=   3
**** TIME= 0.8502E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2588E-03  LOOPS=   3
**** TIME= 0.8668E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2567E-03  LOOPS=   3
**** TIME= 0.8835E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2540E-03  LOOPS=   3
**** TIME= 0.9002E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2533E-03  LOOPS=   3
**** TIME= 0.9168E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2515E-03  LOOPS=   3
**** TIME= 0.9335E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2499E-03  LOOPS=   3
**** TIME= 0.9502E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2481E-03  LOOPS=   3
**** TIME= 0.9669E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2465E-03  LOOPS=   3
**** TIME= 0.9835E+00 HRS  RELX= 0.5000E-03  RXCC= 0.2450E-03  LOOPS=   3
**** TIME= 0.1000E+01 HRS  RELX= 0.5000E-03  RXCC= 0.2430E-03  LOOPS=   3
**** SIMULATION COMPLETED: PERFORMING WRAP-UP...

CASEA_MB6_1 >
```

Note that the number of loops or iterations required to obtain convergence is large for the first few time steps. This is normal since it takes a few time steps for the pressure profile to settle and converge on the final values. Once the pressure profile has settled, only a few loops/iterations are required to reach convergence, hence the decreasing number of loops as the simulation progresses. The convergence value is specified in the control data base (as shown in fig. 12). Convergence is reached when the difference between the current and previous time steps for all parameters (temperature, pressure, etc.) is less than this specified convergence value.

Upon completion of the simulation, the user may examine the results at each output interval by accessing the MB6_1.LPP file. This file may be accessed from CASE/A with the EDT command. A segment of the MB6_1.LPP file for this example case is shown in figure 13.

Data plots of simulation results may also be obtained through the IPU, which will be explained later in sections 4.3 and 5.4 of this manual. An example plot from this case is shown in figure 14. This plot shows the LABA air temperature in degrees Fahrenheit versus time simulated for the 1-h run in this section. Figure 15 shows another plot from the same case but from a longer simulation run of 48 h.

After viewing the output data and extracting information from the case, the user can eliminate the case with the DELCASE command. To do this enter the following:

```
DELCASE
```

CASEA responds with the following prompt and information when entering this command:

```
CASEA_MB6_1 > DELCASE
Are you SURE you want to DELETE Active Case? (Y/[N]) > Y
CASE DELETION COMPLETE (DATA BASE ENTRIES ONLY).
USER MUST DELETE ALL OTHER ASSOCIATED MODEL FILES:
NAME.MOD,  NAME.CMP,  NAME.FMP,  NAME.LPP,  NAME.FOR
CASEA >
CASEA > EXIT
FORTRAN STOP
```

```
|*********************************************************************
|
|TIME =       0.000000 HRS (    0.000  DAYS)
|STEP =       0.016670 HRS
|RELXX= 0.0005000
|RELCC= 0.0000000
|LOOPS=         0
|
|.....................................................................
|...................     SUBSYSTEM:  LAYOUT        ...................
|.....................................................................
|
|=====================================================================
|CASE: MB6_1      COMPONENT: ORBITER    TYPE: CABIN
|RELATIVE NUM:      1                 SUBSYS: LAYOUT
|
|                       ACCUM
|CONSTITUENTS
| TOTAL MASS     LBM       118.67
|  NITROGEN     FRACT   0.6765711
|  CARBON DIOX  FRACT   0.0158325
|  WATER        FRACT   0.0154205
|FLUID PROPERTIES
|  PRESSURE               10.2000
|  TEMPERATURE            80.0000
|  SPEC HT, CP             0.2431
|  DENSITY                 0.0515
|  VISCOSITY               0.0454
|  SPEC HT, CV             0.1746
|RELAXATION VALUES
|  MASS                 0.0000000
|  TEMPERATURE          0.0000000
|  PRESSURE             0.0000000
|=====================================================================
|CASE: MB6_1      COMPONENT: PBM1       TYPE: CABIN
|RELATIVE NUM:      2                 SUBSYS: LAYOUT
|
|                       ACCUM
|CONSTITUENTS
|  OXYGEN       FRACT   0.2921759
|  NITROGEN     FRACT   0.6765711
|  CARBON DIOX  FRACT   0.0158325
|  WATER        FRACT   0.0154205
|FLUID PROPERTIES
|  PRESSURE               10.2000
|  TEMPERATURE            80.0000
|  SPEC HT, CP             0.2431
|  DENSITY                 0.0515
|  VISCOSITY               0.0454
|  SPEC HT, CV             0.1746
|RELAXATION VALUES
|  MASS                 0.0000000
|  TEMPERATURE          0.0000000
|  PRESSURE             0.0000000
|=====================================================================
```

Figure 13. Introductory example problem partial .LPP file.

The user may then exit CASE/A and delete the files mentioned above using the VAX/VMS DELETE command.

These introductory example models have shown the basic procedures required to generate and execute a CASE/A simulation. The emphasis was on program essentials and no attempt was made to cover program details. For instance, the operations logic required to timeline the cold plate heat loads and control the flow control valve positions was not discussed. A more detailed explanation of operations logic can be found in section 7.0 and in the water conditioning model sample problem, located in appendix A.

1.4 Manual Organization

This section of the user's manual has presented the basic concepts and development history of the CASE/A program. In addition, a brief overview of the program and its capabilities were discussed, and two simple introductory problems were presented to familiarize the reader with CASE/A and its usage.

Section 2.0 presents an overview of the schematic management commands available within the CASE/A program. Topics discussed are the creation and deletion of a case, location of components on a subsystem screen, connection of

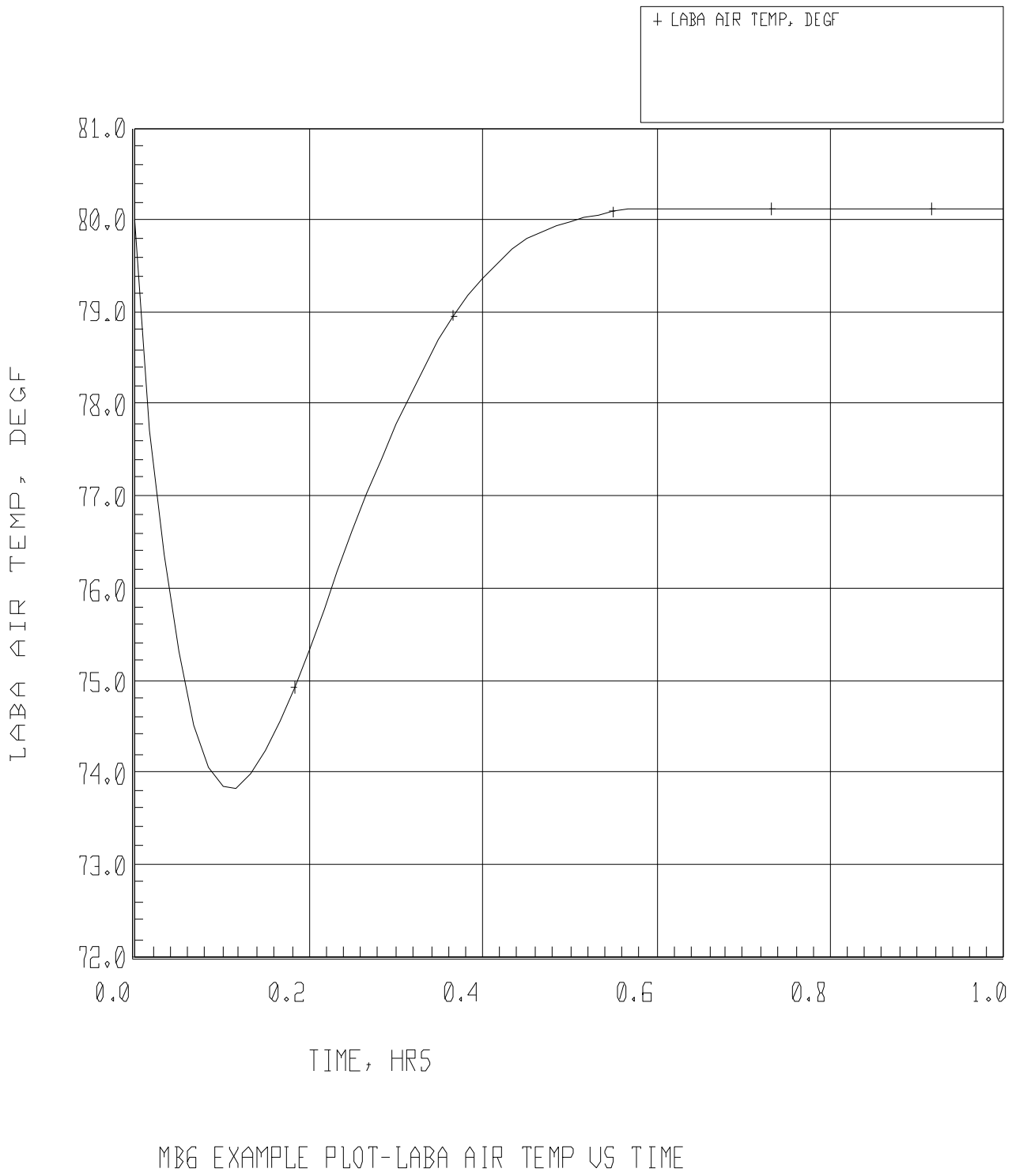


Figure 14. MB6_1 example CASE/A output plot from 1-h run.

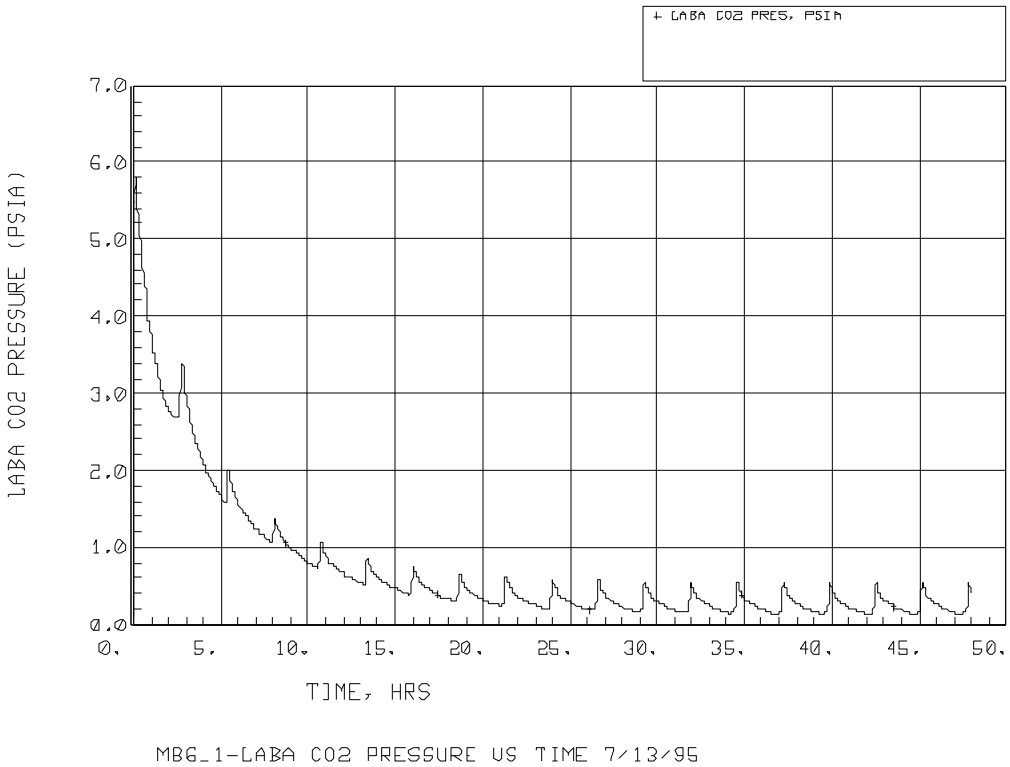


Figure 15. MB6_1 example CASE/A output plot from 48-h run.

the components with fluid streams, and manipulation of the component icons on the screen.

Section 3.0 describes the CASE/A component data base management system. The data base architecture is discussed along with the full screen data base editing features which allow the user to access and edit data for each component located in their model. The data base line editor is also briefly discussed for those users without full screen editing capability.

Section 4.0 describes the simulation control and execution system of the CASE/A program. The control edit screen is described, which contains the control constants which the user manages to control the simulation. Among the control constants are start time, stop time, maximum number of system iteration loops, and system convergence criteria. The chemical constituents label edit screen allows the user to update the list of fluid stream constituents to be tracked in a simulation. The IPU plot data setup screen is described, which allows the user to define the variables that will be written to a data file during the simulation for subsequent plotting after the simulation. The canned plotset utility is described, which allows the user to quickly set up plots from plotset templates previously created and tested. The last command described in this section is the `SOLVE` command, which is used to initialize a simulation solution.

Section 5.0 describes the model output management features of the program. This section describes how to generate hard copy output of subsystem schematics and component data bases. The tabular summary `.LPP` file, the IPU commands, custom user output, and schematic connection and hydraulic flow maps are also discussed in this section.

Section 6.0 describes the utility commands provided with the CASE/A program. These system utility commands enable the user to transfer subsystem screens between CASE/A models, change VAX terminal settings, assign subsystem heat loads to a particular cabin environment, request on-line help information, and temporarily exit from CASE/A to the VAX operating system or editor. In addition, a

description is provided for the process to archive CASE/A models.

Section 7.0 describes the user operations FORTRAN logic that can be used to custom control a CASE/A simulation. There are seven points within a CASE/A simulation where the user may intervene in the solution process to change component variables. The user may also sequence the calling order of the components in order to speed the solution process. The timer, controller, and blackbox system-provided components are described, which enable the user to timeline, control, and customize component functions.

Section 8.0 describes the analytical techniques used to model the physical/chemical component processes. The pressure drop solution methodology is described along with heat and mass transfer network modeling techniques.

Section 9.0 provides important tips for the new CASE/A user. This section describes common mistakes and things to avoid when constructing a CASE/A model. In addition, cautions are provided on common user problems with the hydraulic network setup and solution convergence process.

Section 10.0 provides a complete description of each component currently included in the CASE/A component library. These descriptions include the details of the modeled process, the component icon layout, and the component edit screen(s) with the default data included.

Appendix A includes two sample problems which take the user step-by-step through the CASE/A modeling process; from initial schematic construction to model output generation.

Appendix B provides the user with an alphabetized quick-reference section of all CASE/A commands, including command descriptions and formats.

Appendix C provides installation notes to help new users install CASE/A on their machines.

1.5 References

1. "G-189A Generalized Environmental/ Thermal Control and Life Support System Analysis Computer Program," 1977.

2. Ishimoto, T., and Fink, L.C.: "Systems Improved Numerical Differencing Analyzer," 1971.

3. "PROCESS Simulation Program," Simulation Sciences Inc., Fullerton, California.

4. "CHEMCAD," COADE Engineering Software, Houston, Texas.

5. "Computer Aided System Engineering and Analysis (CASE/A) Programmer's Manual-ECLSS Series, Version 4.1." McDonnell Douglas Space Systems Company-Huntsville Division, MDC W5146-4, October 1990.

6. Knox, J.C. and Gilley, S.D.: "Evaluation of the Carbon Dioxide Removal Assembly Requirements for the Space Station *Freedom* in the Manned Tended Capability Through Permanently Manned Capability Configurations," 1992.

2.0 SCHEMATIC MANAGEMENT

This section describes the CASE/A commands available to generate schematic representations of actual systems to be modeled. These commands will generally be used to drive graphics subroutines located in the CASE/A library. This section includes a discussion on how to initiate the CASE/A program and how to terminate a session. Commands are described that allow the user to create an initial model case and load, duplicate, save, or delete existing cases. Schematic manipulation commands are also described that allow the user to create subsystem screens and to locate components and connections on the screens. In addition, a variety of commands are described that enable manipulation of the components and connections to improve the layout and appearance of the subsystem screen. It is advisable to enter CASE/A commands in ALL CAPS to assure recognition by the command processor. The semicolon delimiter ";" is optional and may be substituted with a blank space.

2.1 Program Initiation and Termination

The CASE/A modeling system typically resides in several different directories on the VAX computer. For instance, the FORTRAN code resides in the `[CASEA.CODE]` directory. The component data bases reside in the `[CASEA.DATA]` directory. The user directory is normally named `[user root name.CASEA]`. It is recommended that the user execute CASE/A from the user directory. To do this, the `LINKCAS.COM` file must be located in the user directory. A more detailed explanation on CASE/A program installation is provided in appendix C.

The user must first link with the CASE/A program. This is accomplished with the following command entered from the user directory:

```
$ @LINKCAS
```

This command does not have to be repeated unless a change is made to the main CASE/A program.

If optional operations logic is used, it is recommended that the FORTRAN file be named the same as the case name (`CASENAME.FOR`). After compiling the OPS logic FORTRAN file (see section 7.5), the user then links with CASE/A as follows:

```
$ @LINKCAS CASENAME
```

This command will have to be repeated each time `CASENAME.FOR` is recompiled to incorporate changes in OPS logic code.

The `LINKCAS.COM` file assumes that two `ASSIGN` statements for the logicals `CASEA$CODE and CASEA$DATA` are included in the user's `LOGIN.COM` file. The current version of `LINKCAS.COM` links with the VAX debugger program. If the user desires to run the program with the debugger, then the following command is issued:

```
$ RUN CASEA
```

To run the program without the debugger, the user types:

```
$ RUN/NODEB CASEA
```

After executing the program, the user will enter the top-level CASE/A environment denoted by the "`CASEA >`" prompt. The user is now ready to start a new case or load an existing case with the commands described in section 2.2. These subsequent commands will take the user into a particular CASE/A model case environment denoted by a "`CASEA_CASENAME >`" prompt.

To terminate a CASE/A session, the user issues the `EXIT` command.

<u>EXIT</u>

The `EXIT` command will terminate a CASE/A session and automatically save the current model configuration to the CASENAME.MOD file. The user will be returned to the DCL "`$`" prompt.

Format: `EXIT`

2.2 Model Creation, Loading, and Deletion

This section describes the commands required to create a new CASE/A model case as well as to load, duplicate, save, and delete an existing case.

2.2.1 NEWCASE

The `NEWCASE` command will initialize a new case. The user may then begin the process of naming subsystems and locating components and connections. If the user is already in an existing case, this command will switch the user from the current case to the new case. The current case will be automatically saved before entering the new case.

Format: `NEWCASE;CASENAME`

Note: This command may be invoked from either the "`CASEA >`" system prompt or the "`CASEA_CASENAME >`" case prompt.

2.2.2 LOADCASE

The `LOADCASE` command will load an existing case into memory. This command is used to load an existing case only. If the user is already in an existing case, this command will switch the user from the current case to the requested second case. The current case will be automatically saved before loading the new case.

Format: `LOADCASE;CASENAME`

Note: This command may be invoked from either the "`CASEA >`" system prompt or the "`CASEA_CASENAME >`" case prompt.

2.2.3 SAVEAS

The `SAVEAS` command will duplicate an existing case in its entirety and save it under a new name. The user must be in an existing case to invoke this command. The old case will be automatically saved before entering the new case.

Format: `SAVEAS;NEWCASENAME`

2.2.4 SAVE

The `SAVE` command will save the current case to the `CASENAME.MOD` file. This is performed automatically following a component location (`LO`) command and during the file wrap up for the `EXIT` command.

Format: `SAVE`

2.2.5 DELCASE

The `DELCASE` (delete case) command will delete an existing case. All data base records associated with the case will be deleted. The user must delete the `CASENAME.MOD` file and any OPS logic FORTRAN file associated with the case from the DCL environment after leaving CASE/A.

Format: `DELCASE`

Note: This command must be invoked from the "`CASEA_CASENAME >`" case prompt. The current active case will then be deleted.

2.2.6 DIR

The `DIR` (directory) command allows the user to identify all subsystems and components associated with a specific case. When this command is invoked, each subsystem name along with the components and their respective names are displayed on the screen.

Format: `DIR`

2.3 Schematic Manipulation

The following three sections describe the commands required to create subsystem screens and to locate components and connections. Manipulation commands that pertain to one or more component icons, such as a grouped move command, are described in section 2.3.1. Similar commands that only operate on a single component icon are described in section 2.3.2.

2.3.1 Subsystem Screen Layout

This section describes the commands associated with subsystem screen creation and appearance. These commands allow the user to create and manipulate component icons, notes, and labels on the subsystem screens.

a. SS

The SS (subsystem) command allows the user to define the subsystem within the current case upon which subsequent operations are to be performed such as adding, moving, or deleting components or establishing or modifying connections. Components on different subsystem screens may be connected with the bubble component feature described in section 10.

Format: SS;SSNAME

Notes: The SSNAME field is limited to eight characters. An automatic redraw (RD) is executed upon issuance of the SS command.

b. RD and RD+

The RD (redraw) command is used to refresh the current subsystem screen. This may be invoked at any time to clear unwanted messages from the screen and to show the effect of component moves and deletions. RD+ (redraw plus) is used with the LAB command so that the labels generated are included on the screen. Component stream connection hit boxes are omitted with the use of the RD+ command.

Format: RD or RD+

c. M

The M (move) command allows the user to move a portion (or all) of a subsystem screen to a new position on the subsystem screen by drawing a box around the components to be moved. Connections and intermediate points of connections inside the box will also be moved. Any notes and labels within the box are also moved. After the command is entered, the crosshairs (cursor) will appear three times. The first appearance allows the user to designate one of the box corners by typing any character. The second appearance of the cursor allows the user to designate the other box corner (diagonally opposing the first corner) by typing any character. After the second box corner is determined, the box is drawn on the subsystem screen with a small marker in the center of the box. The third appearance of the cursor allows the user to designate the location of the box "center-marker" at the new subsystem screen location by typing the character "C" for continue. If any other character is typed at this point, the M command is terminated.

Format: M

Notes: The maximum number of components that can be grouped and moved is 50.

d. COPYALL

The COPYALL command will allow the user to copy a portion (or all) of a subsystem screen to a new or existing subsystem screen by drawing a box around the components to be copied. The command argument is the destination subsystem. After the command is entered, the crosshairs (cursor) will appear three times. The first appearance allows the user to designate one of the box corners by typing any character. The second appearance of the cursor allows the user to designate the other box corner (diagonally opposing the first corner) by typing any character. After the second box corner is determined, the box is drawn on the subsystem screen with a small marker in the center of the box. The third appearance of the cursor is used to designate the location of the box "center-marker" on the object (destination) subsystem. The user moves the crosshairs to the new location and types the character "C" for continue. If the box is to appear in the same location on the object screen as it appears in the original subsystem screen, then the character "D" (for default) is typed. If any other character is typed at this point, the COPYALL command is terminated. Connections are copied, if both endpoints are on components inside the box. Any notes and labels within the box are also copied. The user is prompted for new equipment names for the copied components (eight characters maximum). A warning will appear on the screen if the new name is a duplicate of any other component name in the case.

Format: `COPYALL;SSNAME`

Notes: The maximum number of components that can be grouped and copied is 50. The BLACKBOX component cannot be copied.

e. MERGE

The `MERGE` command allows the user to combine a complete subsystem from another case. This is especially helpful when developing large models. Complete subsystems can be built and verified as separate cases and then the subsystem merged into the larger model. Development time is reduced since the models are developed as smaller, more manageable problems and then assembled into a single model. The routine will check for duplicate component names and solicit a new name from the user. The new name entered will not be checked for duplicates. If the user does not want to change the name, a carriage return is entered and execution will continue. A recent addition to this command is the ability to merge a case from an older version of CASE/A into a newer version of CASE/A.

The `MERGE` command may also be used to move subsystems or cases across VAX directories or hosts. A complete description of this `MERGE` command function can be found in section 6.6.

Format: `MERGE;CASENAME;SSNAME`

Notes: If the case name or subsystem name are in error or are omitted, the program will solicit a new input.

f. D

The `D` (delete) command allows the user to delete a portion or all of a subsystem screen by drawing a box around the components to be deleted. After the command is entered, the crosshairs will appear two times. The first appearance allows the user to designate one of the box corners by typing any character. The second appearance of the cursor allows the user to designate a box corner diagonally opposed to the first corner and the box will be drawn around the components to be deleted. The user then types the character "D" to delete the selected components or any other character to abort the command and return to the command prompt. The deleted components will disappear on the next subsystem `RD` (redraw) command.

Format: `D`

g. NOTE

The `NOTE` command allows the user to place a note on a subsystem screen. The note appears immediately after completion of the `NOTE` command and will appear with the subsystem whenever the subsystem is displayed. After the command has been initiated, the screen crosshairs (cursor) are activated and the user can place the note in the desired location by typing any character. The required command arguments are shown in the format definition below. `NT` determines the string output format and the allowable options for `NT` are as follows:

`L` – Each string is left-justified from the cursor location, starts as a new line, and can contain a maximum of 20 characters.

`C` – Each string is centered over the cursor location, starts as a new line, and can contain a maximum of 20 characters.

`N` – Only one string is allowed, it is left-justified from the cursor location, and can contain a maximum of 80 characters.

`CSZ` determines the character size of the note strings and the allowable options for `CSZ` are: 1, 2, 3, or 4, where 4 designates the smallest character size.

Format: `NOTE;NT;CSZ;STRING1;STRING2;STRING3;STRING4`

Caution: The semicolon delimiter should always be used with this command to preserve blank spaces in the note itself.

Example: NOTE;L;4;This is note 1;This is note 2;This is note 3;

h. MVNOTE

The `MVNOTE` (move note) command allows the user to move an existing note to relieve

layout congestion or otherwise improve the appearance of the subsystem screen. The crosshairs are actuated, and the user designates a note by placing the crosshairs on the note and typing any character. The crosshairs remain active, and the user designates a new location for the note by again typing any character. The note will appear in the new location immediately, and the old note will disappear on the next RD command.

Format: `MVNOTE`

i. DELNOTE

The `DELNOTE` (delete note) command allows the user to delete an existing note from a subsystem screen. The crosshairs are activated, and the user designates the note to be deleted by placing the crosshairs on the note and typing any character. The note will disappear on the next `RD` (redraw) of the subsystem.

Format: `DELNOTE`

j. LAB

The `LAB` (label) command allows the user to display stream or component characteristics on the subsystem schematic. This command is normally used after a system solution. To initiate this command, three arguments are required, as shown in the format below. `NC` represents the desired constituent, property, or data base parameter as defined in table 7 for the constituent array, table 8 for the PROperty array, or the solution routine overviews (section 10) for the component data base CON array. `LABEL` is the title to be printed out with the value and is limited to eight characters. `INA` represents which array is being referenced, 1 for constituent array, 2 for property array, and 3 for CON array. Once the command has been activated, the crosshairs will appear, whereupon the user can specify the desired stream and component to which the label applies by placing the crosshairs in the hit box of that stream, or on the component for the CON array values, and typing any character. The crosshairs will again appear to allow the user to specify the desired location of the label by typing any character with the crosshairs at the desired location.

Format: `LAB;NC;LABEL;INA`

k. MVLAB

The `MVLAB` (move label) command allows the user to move an existing label to relieve congestion or otherwise improve the subsystem appearance. The crosshairs are actuated and the user designates a label by placing the crosshairs on the label and typing any character. The user designates a new location for the label by moving the crosshairs and typing any character. The label will appear in the new location immediately, and the old label will disappear after a RD command.

Format: `MVLAB`

l. DELLAB

The `DELLAB` (delete label) command is used to delete an existing label. The crosshairs are activated, and the user designates the label to be deleted by placing the crosshairs on the label and typing any character. The label will disappear on the next `RD+` (redraw plus) command.

Format: `DELLAB`

m. RNSS

The `RNSS` (rename subsystem) command will allow the user to change the name of the subsystem screen that is currently activated. The command argument is the new subsystem name. The new subsystem name will appear upon the next `RD` (redraw) of the subsystem.

Format: `RNSS;NEWSSNAME`

Notes: All components, notes, labels, and cabin assignments associated with the subject subsystem are updated accordingly.

n. VECT and NOVECT

The `VECT` (vector) command enables the use of vector characters. The `NOVECT` (no vector) command disables the use of vector characters.

Format: `VECT` or `NOVECT`

2.3.2 Component Manipulation

This section describes the commands associated with component location and manipulation. These commands allow the user to locate, delete, rename, and reposition component icons on the subsystem screen.

a. LO

The LO (locate) command is used to locate a new component in the existing subsystem. The crosshairs are activated to allow the user to define a location for the component on the screen. To define a location, the user types any single character with the crosshairs at the desired location. The component figure is then drawn. A unique name for each component is suggested. This will simplify any logic in the OPS blocks that must refer to this component.

Format: LO;EQTYPE;EQNAME

Example: LO;CHX;CHX1 (locates a condensing heat exchanger named "CHX1")

Notes: EQTYPE is the mnemonic associated with the equipment type. A list of the legal equipment types (EQTYPE) is provided in table 10.0-1. The equipment names (EQNAME) may be up to eight characters; however, this may exceed the space available in some component icons.

b. DELEQ

The DELEQ (delete equipment) command is used to delete a component from the existing subsystem along with its connections. The crosshairs are activated and the user designates the component to be deleted by placing the crosshairs within the figure and typing any character. The component and all its connections will disappear on the next RD (redraw) command.

Format: DELEQ

c. RNEQ

The RNEQ (rename equipment) command allows the user to change the name of a component on a subsystem screen. The second argument of the command is the new equipment name. After the command is entered, the crosshairs are activated and the user designates the component in the subsystem to be renamed by placing the cursor on that component and typing any character. The new name will appear upon the next subsystem RD (redraw) command.

Format: RNEQ;NEWEQNAME

Notes: The new equipment name can contain a maximum of eight characters. A warning will appear on the screen if the new name is a duplicate of any other component name in the case.

d. MV

The MV (move) command allows the user to move an existing component to relieve congestion or otherwise improve the subsystem layout. The crosshairs are actuated, and the user designates a component by placing the crosshairs within the figure and typing any character. The crosshairs remain active, and the user designates a new location by again typing any character. The figure will appear in the new location, and the connections are dragged with it upon the next subsystem RD (redraw) command.

Format: MV

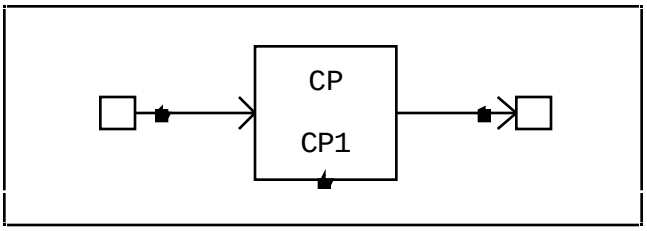
e. ROTATE

The ROTATE command will allow the user to rotate an existing component. Once the command has been activated, the crosshairs will appear, and the user can specify the component to be rotated. This is accomplished by typing any single character with the crosshairs placed on the desired component. The component will be in the new orientation on the next RD (redraw) command. To initiate this command, one argument is required as shown in the format below. ANGLE represents the degree of rotation that is desired. The rotation angle must be a multiple of ±90˚ with a magnitude not greater than 360˚. A positive angle represents a counterclockwise rotation.

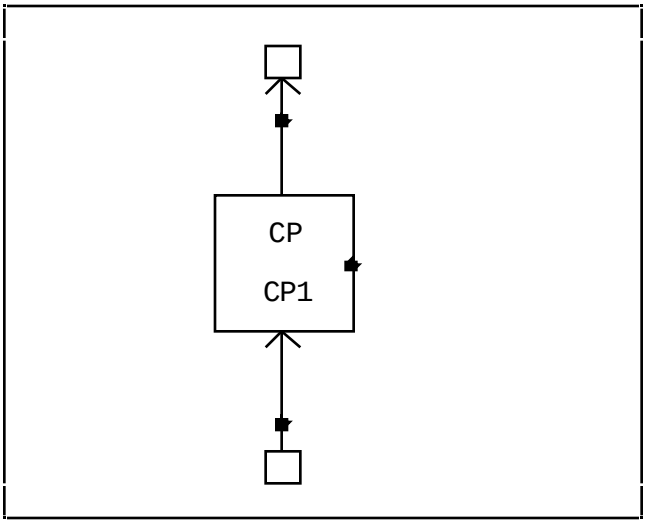
Format: ROTATE;ANGLE

Example: The command LO;CP;CP1 would draw the following default cold plate

component icon orientation on the screen:



A subsequent `ROTATE;90` command would activate the crosshairs on the screen. The user would then place the crosshairs on the component icon to be rotated and type any keyboard character. Upon the next `RD` command, the icon would appear rotated 90˚ counterclockwise as follows:



2.3.3 Stream Connections

This section describes the commands associated with component fluid stream connections. Commands that make connections and delete existing connections are described.

a. CN

The `CN` (connect) command allows the user to specify an interconnect path between components. The crosshairs are activated, and the user specifies a starting point by positioning the crosshairs within the small hit box on the component stream and typing the character "S." The user then defines the path to the next component. If intermediate points are desired (segmented line), the user designates the intermediate point locations by typing the character "I" at the desired location(s). The user designates the termination point by placing the crosshairs within the second component stream "hit box" and typing the letter "E." The path is drawn on the screen as it is defined. If an existing path is redefined, the old path will disappear on the next `RD` (redraw) command. The crosshair movement on some terminal types can be sped up by holding down the shift key in conjunction with the desired cursor key. The two arguments `LEQ` and `DEQ` are optional connection parameters used in the calculation of the pressure drop through the connection. The parameter `LEQ` is the hydraulic equivalent length, in feet, of the connection. The parameter `DEQ` is the hydraulic equivalent diameter, input as hundredths of an inch (i.e., `DEQ` = 100 for an equivalent diameter of 1 inch). The absence of either argument results in a pressure drop of zero for the connection.

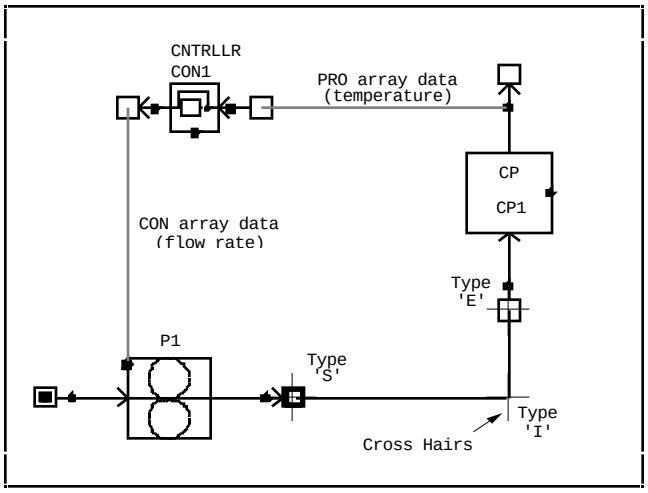
The `CN` command is unique for the CNTRLLR and TIMER components. These components require the user to connect data streams to "hit dots" located either on the component icon body or on the inlet or outlet component streams. The data streams appear as dotted lines on the subsystem screen. See sections 7.6 and 10.0 for complete discussions of the CNTRLLR and TIMER components.

Format: `CN[;LEQ;DEQ]` (fluid streams)
`CN` (data streams)

Notes: The two arguments `LEQ` and `DEQ` for the fluid stream `CN` command are optional; the default pressure drop of a connection is 0.0.

Example: A pump is located on the screen, and it is desired to connect a fluid stream between the pump outlet and the cold plate inlet. To do this, the `CN` command is entered and crosshairs appear on the screen. The crosshairs are placed on the pump outlet stream hit box and any character is typed. The same procedure is followed to locate the intermediate point and the end point as shown in the following.

A CNTRLLR component is located on the screen, and it is desired to control the pump flow rate based on the cold plate outlet stream temperature. The CNTRLLR input data (PRO array) stream is connected between the CNTRLLR input hit box and cold plate outlet stream hit dot. Similarly, a data (pump CON array) stream is connected between the CNTRLLR outlet hit box and the pump icon body hit dot as shown in the following.



b. DELCN

The DELCN (delete connection) command allows the user to delete a connection from the existing subsystem without affecting component placement. The crosshairs are activated and the user designates the starting location of the connection by placing the crosshairs on the component stream hit box and typing any character. The crosshairs remain active and the ending point is designated in a similar manner. The connection will disappear on the next RD (redraw) command.

Format: DELCN

3.0 COMPONENT DATA BASE MANAGEMENT

This section describes the component data base layout. Component data from centralized mass storage is loaded into the CON array during case solution execution. By accessing the component edit screens prior to issuing the SOLVE command, the user is able to change component input data records in mass storage before it is loaded into the CON array for a particular case. For those users without full screen editing capability, a line editor is provided to perform the same editing functions, but with a sacrifice in convenience.

3.1 Component Data Base Layout

Most of the component types currently supported by the solution system allow the user to specify the performance characteristics that are used in the component routines to simulate their operation. Also, many of the component routines record various operating parameters and benchmark values of interest. This type of component parameter data is maintained and edited in mass storage. When a CASE/A model solution is initiated (SOLVE command), the component data records pertaining to the particular case are loaded into the one-dimensional CON array. The CON array data are grouped in component blocks by component identification number (IEQ). A component data block consists of four data location sections as shown in figure 16 and discussed in the following four paragraphs.

The fixed data section always consists of nine parameters for each component. These data locations include (1–3) the number of component inputs, outputs, and benchmarks; (4–6) the maximum values for mass, temperature, and pressure relaxation; and (7–9) the component power, weight, and volume.

The input data section contains the performance input data for the particular component. This data section may vary in length for different components depending on the complexity of the modeled process. An example of a process input parameter is a cold plate heat load.

The output data section may also vary in size depending on the component. These output locations are updated after each time step. An example of a component output parameter is the cold plated equipment temperature.

The benchmark data section is updated once at the end of a simulation. This data section, which also varies in length by component, tracks minimum, maximum, and nominal values of key process parameters. Example benchmark data are the minimum, maximum, and nominal cold plated equipment temperature.

The CON array is initialized at the beginning of a simulation solution by loading the input data from the data base files and setting the output and benchmark data to appropriate values in the component initialization logic segments. The various component routines access these data by use of special location pointers. In addition to the presimulation editing capability, the user may also access and modify the CON array data directly by use of the GETK and SETK routines in the user operations logic blocks (see section 7.2). The CON array is currently sized for 25,000 total parameters.

3.2 Data Base Screen Editor

The component data base screen editor provides a quick, efficient method of changing component input data parameters prior to executing a CASE/A model solution. The typical component edit screen consists of three sections: (1) input, (2) output, and (3) benchmark data. The user has editing capability in the input section only. The output and benchmark sections are provided for postsimulation reference purposes.

An example of a typical component main edit screen is shown in figure 17 with the input section shaded. Key areas of the input data section are noted in the figure. The numbers to the left of each input item are used in the SETK and GETK user operations routines described in section 7.2. The edit screen is invoked with either the E or ED command. The key functions of the editor have been generalized to the maximum extent practical to support a variety of terminals. Because the return codes differ from terminal to terminal for the special keyboard functions such as “page up,” “page down,” and “insert”/“delete” functions, an internal keyboard array is maintained to allow the

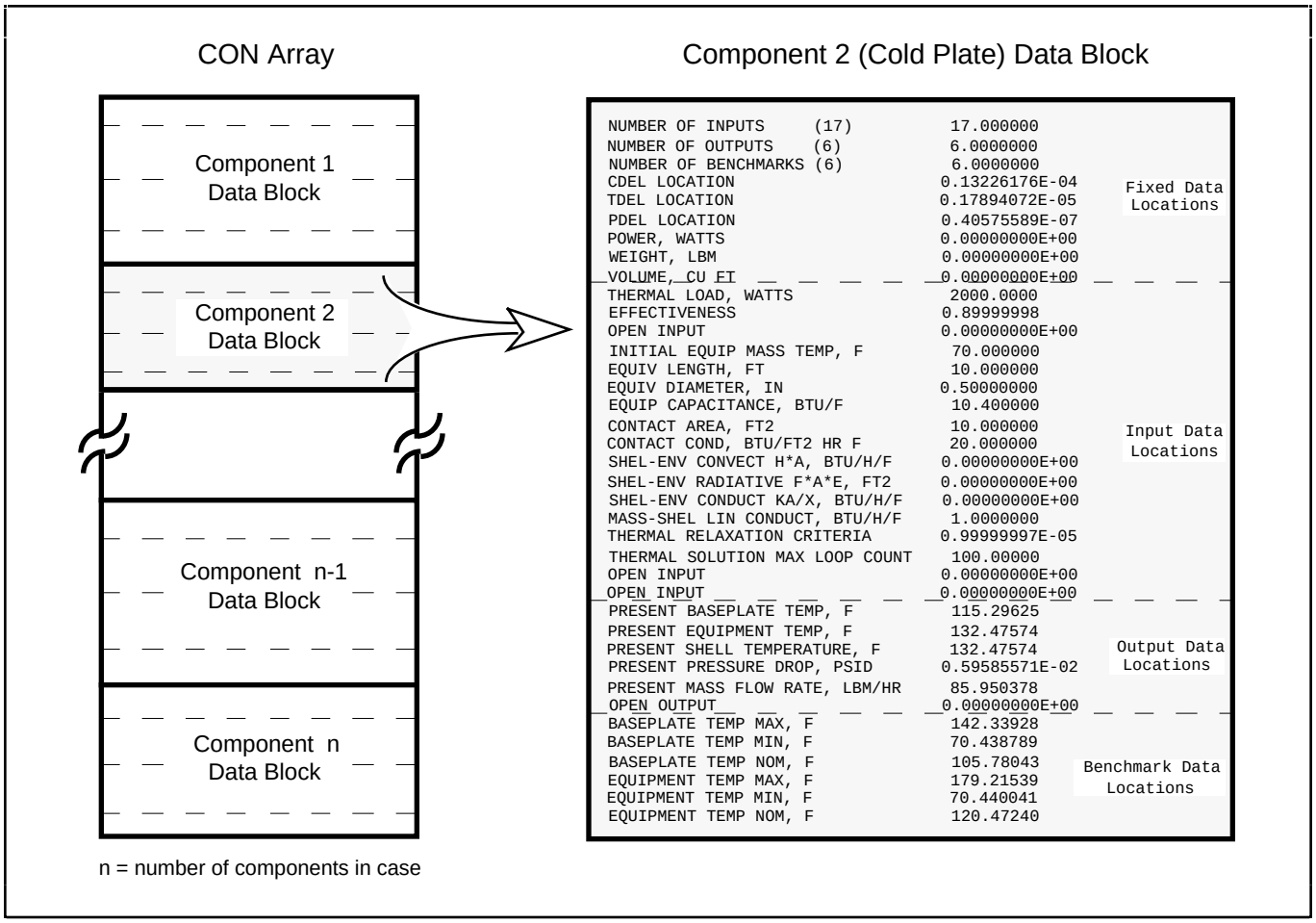


Figure 16. CON array layout.

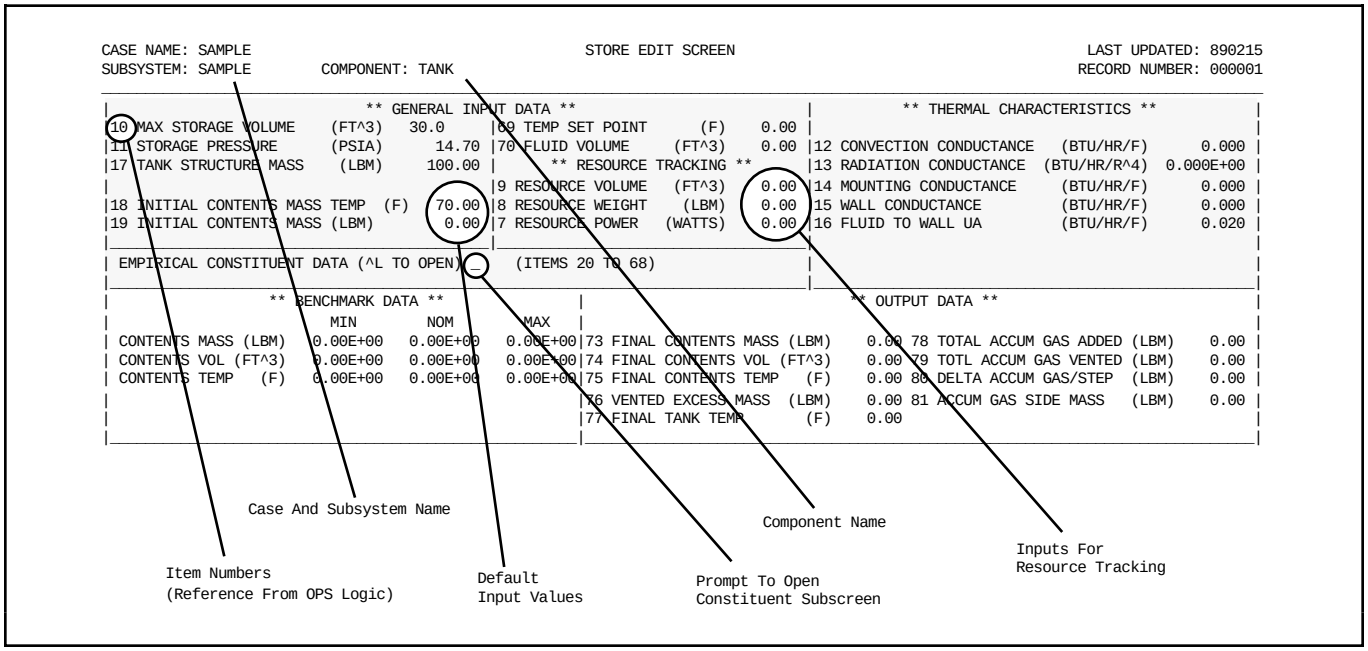


Figure 17. Typical component main edit screen input area.

proper screen action to be taken as a function of the terminal type selected with the TERM command (see section 6.1). To support terminals with no function key/keypad features, all cursor control and function key actions may be initiated by a set of basic control functions (<Ctrl A>, <Ctrl Z>, etc.) which are available to nearly all terminals. A summary of control key functions for the main screen editor is provided in table 2.

Table 2. Main screen editing command functions.

MAIN SCREEN EDITOR

Function	Key	Equivalent
Cursor up	Up arrow key	
Cursor down	Down arrow key	<Ctrl M>
Cursor left	Left arrow key	<Ctrl J>
Cursor right	Right arrow key	<Ctrl K>
Abort	F11	<Ctrl X>
Exit and save	Next screen key	<Ctrl Z>
Character insert	Insert key	<Ctrl A>
Character delete	Remove key	<Ctrl R>
Enter subscreen/ list editor		<Ctrl L>

The editor uses a VT-100 protocol to control the screen and cursor position. The user may jump from item to item using the TAB key or up and down arrow keys. The cursor may be positioned within an item using the left and right arrow keys. The editor is always in overstrike mode, with typed characters replacing the data at the current cursor position. Data may be deleted at a given position in the item by repeatedly using the REMOVE key, resulting in the remainder of the field moving left to fill the position. Space may be inserted into the item by repeatedly using the INSERT HERE key to generate the desired number of spaces at the cursor position (trailing data are pushed right). The spaces may then be filled with data using the overstrike entry.

A summary of the E and ED commands is provided in the following sections.

3.2.1 E

The E (edit with cursor) command allows the user to edit the data base edit screen for a particular component in the system. The user must be on a composite graphics/text terminal (or emulation) with the TERM command set to "3." After the E command is entered, the crosshairs (cursor) are activated and the user designates the component to be edited by placing the cursor on that component and typing any character. The screen editing mode is activated and the arrow keys are used to advance to the desired parameter location and a new value is entered. A <Ctrl Z> saves the data values and exits from the editor. The data are stored permanently for future reference. If the user decides not to save the adjusted values, then the F11 key (or <Ctrl X>) is typed.

Format: E

Notes: Functionally equivalent to the ED command.

3.2.2 ED

The ED (edit without crosshairs) command allows the user to edit the data base edit screen for a particular component in the system. The user may be on a text-only or composite terminal when issuing this command. The TERM command must be set to "2" or "3" depending on which type terminal is being used (see section 6.1 for explanation of TERM command). This command has the same options and syntax of the E command, but the user must indicate the type or name of the component to be edited. If more than one component type is present in a case, then CASE/A prompts the user with a numbered list of the component names of that particular equipment type. The user then enters the number of the component that is desired for editing.

Format: ED;EQTYPE (option 1)
ED;EQNAME (option 2)

Examples: ED;CHX (prompts the user with a list of all CHX components in the case if more than one exists).

ED;LABHX (directly edits the CHX component name LABHX).

Notes: EQTYPE is the mnemonic which is associated with the equipment type (see table 10.0-1). EQNAME is the name given to the component by the user.

3.2.3 FASTED

The `FASTED` (fast edit) command allows the user to change, with a single command, input parameters for all components of a particular type (i.e., CP, PUMP, CHX, etc.) in a case. The edit screen of the particular component type that was located first during subsystem screen construction will appear. As changes are made to this screen, they will automatically be made for the same item number in all the other records of that component type in the case. It is important to note that individual item numbers only are changed throughout the case; carbon copies of the edit screen are not made in order to protect other user data.

Format: `FASTED;EQTYPE`

Example: `FASTED;CP` (allows the user to change selected items for all the CP (cold plate) components in a case).

Notes: Due to the limitations of the overstrike method of editing component edit screens, the item changed must differ from the previous value in some way to assure changes in all the other data base records. For example, if an item value is 10.0 in the edit screen that first comes up, the user may not change all other similar items to 10.0 by simply typing 10.0 over the existing number. The editor will not recognize the change. The user may, however, change the item to 10.1 and then back to 10.0 to assure changes are made across the board.

The user may only edit input data values shown in the gray-shaded region of figure 17. The case, subsystem, and component names may not be accessed in the edit screen. Other commands exist to perform these name change operations (see section 2.3).

Many of the CASE/A component routines require subordinate screens due to the large number of data parameters required to model the given process. This feature allows a secondary screen to be invoked to permit the editing of data subsets that either will not fit on the main screen, are infrequently edited, or in the opinion of the component routine developer, logically belong in a separate display. Any number of subordinate screens may be accessed from the main screen but a subordinate may not have additional subordinates. A field is established on the main screen to serve as the entry point to the subordinate screen (fig. 17). The user enters the subordinate screen by positioning the cursor on the entry point field and typing `<Ctrl L>`. In this example, the empirical constituent data subordinate screen for the STORE component is opened as shown in figure 18. Once in this subscreen, the user manipulates the cursor with the arrow keys to make desired changes to the data. The subordinate screen is opened, and the data may be viewed or

```
CASE NAME: SAMPLE          STORE CONSTITUENT SCREEN        LAST UPDATED: 890215
SUBSYSTEM: SAMPLE           COMPONENT: TANK               RECORD NUMBER: 000001
```

CONSTITNT LABEL	INITL MASS FRACTION	CONSTITNT LABEL	INITL MASS FRACTION	CONSTITNT LABEL	INITL MASS FRACTION
CONSTIT 2	0.0000000	CONSTIT 19	0.0000000	CONSTIT 36	0.0000000
CONSTIT 3	0.0000000	CONSTIT 20	0.0000000	CONSTIT 37	0.0000000
CONSTIT 4	0.0000000	CONSTIT 21	0.0000000	CONSTIT 38	0.0000000
CONSTIT 5	0.0000000	CONSTIT 22	0.0000000	CONSTIT 39	0.0000000
CONSTIT 6	0.0000000	CONSTIT 23	0.0000000	CONSTIT 40	0.0000000
CONSTIT 7	0.0000000	CONSTIT 24	0.0000000	CONSTIT 41	0.0000000
CONSTIT 8	0.0000000	CONSTIT 25	0.0000000	CONSTIT 42	0.0000000
CONSTIT 9	0.0000000	CONSTIT 26	0.0000000	CONSTIT 43	0.0000000
CONSTIT 10	0.0000000	CONSTIT 27	0.0000000	CONSTIT 44	0.0000000
CONSTIT 11	0.0000000	CONSTIT 28	0.0000000	CONSTIT 45	0.0000000
CONSTIT 12	0.0000000	CONSTIT 29	0.0000000	CONSTIT 46	0.0000000
CONSTIT 13	0.0000000	CONSTIT 30	0.0000000	CONSTIT 47	0.0000000
CONSTIT 14	0.0000000	CONSTIT 31	0.0000000	CONSTIT 48	0.0000000
CONSTIT 15	0.0000000	CONSTIT 32	0.0000000	CONSTIT 49	0.0000000
CONSTIT 16	0.0000000	CONSTIT 33	0.0000000	CONSTIT 50	0.0000000
CONSTIT 17	0.0000000	CONSTIT 34	0.0000000		
CONSTIT 18	0.0000000	CONSTIT 35	0.0000000		

Figure 18. Typical component constituent edit screen.

```
CASE NAME: SAMPLE                                    STORE EDIT SCREEN                                    LAST UPDATED: 890215
SUBSYSTEM: SAMPLE        COMPONENT: TANK                                                                  RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                            ** GENERAL INPUT DATA **                          |          ** THERMAL CHARACTERISTICS **          |
|10 MAX STORAGE VOLUME    (FT^3)   30.0     |69 TEMP SET POINT      (F)    0.00 |                                                 |
|11 STORAGE PRESSURE      (PSIA)      14.70 |70 FLUID VOLUME     (FT^3)    0.00 |12 CONVECTION CONDUCTANCE   (BTU/HR/F)      0.000 |
|17 TANK STRUCTURE MASS    (LBM)     100.00 |      ** RESOURCE TRACKING **      |13 RADIATION CONDUCTANCE  (BTU/HR/R^4)  0.000E+00 |
|                                           |9 RESOURCE VOLUME   (FT^3)    0.00 |14 MOUNTING CONDUCTANCE     (BTU/HR/F)      0.000 |
|18 INITIAL CONTENTS MASS TEMP  (F)   70.00 |8 RESOURCE WEIGHT    (LBM)    0.00 |15 WALL CONDUCTANCE         (BTU/HR/F)      0.000 |
|19 INITIAL CONTENTS MASS (LBM)        0.00 |7 RESOURCE POWER   (WATTS)    0.00 |16 FLUID TO WALL UA         (BTU/HR/F)      0.020 |
|___________________________________________|___________________________________|                                                 |
| EMPIRICAL CONSTITUENT DATA (^L TO OPEN) _    (ITEMS 20 TO 68)                 |                                                 |
|___________________________________________________________________________________|_________________________________________________|
|                  ** BENCHMARK DATA **                |                              ** OUTPUT DATA **                              |
|                        MIN        NOM        MAX    |                                                                                |
| CONTENTS MASS (LBM)   0.00E+00   0.00E+00   0.00E+00|73 FINAL CONTENTS MASS (LBM)    0.00 78 TOTAL ACCUM GAS ADDED (LBM)    0.00 |
| CONTENTS VOL (FT^3)   0.00E+00   0.00E+00   0.00E+00|74 FINAL CONTENTS VOL (FT^3)    0.00 79 TOTL ACCUM GAS VENTED (LBM)    0.00 |
| CONTENTS TEMP   (F)   0.00E+00   0.00E+00   0.00E+00|75 FINAL CONTENTS TEMP   (F)    0.00 80 DELTA ACCUM GAS/STEP  (LBM)    0.00 |
|                                                     |76 VENTED EXCESS MASS  (LBM)    0.00 81 ACCUM GAS SIDE MASS   (LBM)    0.00 |
|                                                     |77 FINAL TANK TEMP       (F)    0.00                                        |
|_____________________________________________________|_________________________________________________________________________________|
```

Figure 19. Typical component main edit screen output area.

edited. The user returns to the main screen by typing `<Ctrl Z>` or `NEXT SCREEN`. A summary of control key functions for the subordinate screen editor is provided in table 3.

Component output data are written to the edit screen as shown in the gray-shaded region of figure 19. However, the user is not allowed to edit this portion of the screen. These data are provided mainly for postsimulation reference purposes.

Table 3. Subordinate screen editing command functions.

SUBORDINATE SCREEN EDITOR

Function	Key	Equivalent
Cursor up	Up arrow key	
Cursor down	Down arrow key	`<Ctrl M>`
Cursor left	Left arrow key	`<Ctrl J>`
Cursor right	Right arrow key	`<Ctrl K>`
Abort	F11	`<Ctrl X>`
Character insert	Insert key	`<Ctrl A>`
Character delete	Remove key	`<Ctrl R>`
Return to main and save changes	Next screen key	`<Ctrl Z>`

Component benchmark data are also written to the edit screen as shown in the gray-shaded region of figure 20. As with the output section, the user is not allowed to edit data in this section.

In addition to the main screen editor, a scrollable (unlimited length) list editor is provided for IPU functions. A summary of the cursor control functions for plotset editing are provided in table 2. A listing of plotset subscreen-specific cursor control functions is provided in table 4.

Table 4. Plotset subscreen editing command functions.

PLOTSET SUBSCREEN LIST EDITOR

Function	Key	Equivalent
Cursor up	Up arrow key	
Cursor down	Down arrow key	`<Ctrl M>`
Cursor left	Left arrow key	`<Ctrl J>`
Cursor right	Right arrow key	`<Ctrl K>`
Next data field	Tab Key	`<Ctrl I>`
Abort	F11	`<Ctrl X>`
Next screen	Next screen key	`<Ctrl N>`
Previous screen	Previous screen key	`<Ctrl P>`
Character insert	Insert key	`<Ctrl A>`
Character delete	Remove key	`<Ctrl R>`
Insert line	PF2	`<Ctrl V>`
Add line to end	PF3	`<Ctrl B>`
Delete line	PF4	`<Ctrl D>`
Return to main and save changes		`<Ctrl Z>`

```
CASE NAME: SAMPLE                                    STORE EDIT SCREEN                                    LAST UPDATED: 890215
SUBSYSTEM: SAMPLE        COMPONENT: TANK                                                                RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                          ** GENERAL INPUT DATA **                          |          ** THERMAL CHARACTERISTICS **          |
|10 MAX STORAGE VOLUME    (FT^3)   30.0     |69 TEMP SET POINT      (F)    0.00 |                                                 |
|11 STORAGE PRESSURE      (PSIA)      14.70 |70 FLUID VOLUME     (FT^3)    0.00 |12 CONVECTION CONDUCTANCE   (BTU/HR/F)      0.000 |
|17 TANK STRUCTURE MASS    (LBM)     100.00 |      ** RESOURCE TRACKING **      |13 RADIATION CONDUCTANCE  (BTU/HR/R^4)  0.000E+00 |
|                                           |9 RESOURCE VOLUME   (FT^3)    0.00 |14 MOUNTING CONDUCTANCE     (BTU/HR/F)      0.000 |
|18 INITIAL CONTENTS MASS TEMP  (F)   70.00 |8 RESOURCE WEIGHT    (LBM)    0.00 |15 WALL CONDUCTANCE         (BTU/HR/F)      0.000 |
|19 INITIAL CONTENTS MASS (LBM)        0.00 |7 RESOURCE POWER   (WATTS)    0.00 |16 FLUID TO WALL UA         (BTU/HR/F)      0.020 |
|_____________________________________________|_____________________________________|                                                 |
| EMPIRICAL CONSTITUENT DATA (^L TO OPEN) _    (ITEMS 20 TO 68)                 |                                                 |
|_____________________________________________________________________________________|_________________________________________________|
|                  ** BENCHMARK DATA **                |                          ** OUTPUT DATA **                              |
|                        MIN        NOM        MAX    |                                                                                |
| CONTENTS MASS (LBM)   0.00E+00   0.00E+00   0.00E+00|73 FINAL CONTENTS MASS (LBM)    0.00 78 TOTAL ACCUM GAS ADDED (LBM)    0.00 |
| CONTENTS VOL (FT^3)   0.00E+00   0.00E+00   0.00E+00|74 FINAL CONTENTS VOL (FT^3)    0.00 79 TOTL ACCUM GAS VENTED (LBM)    0.00 |
| CONTENTS TEMP   (F)   0.00E+00   0.00E+00   0.00E+00|75 FINAL CONTENTS TEMP   (F)    0.00 80 DELTA ACCUM GAS/STEP  (LBM)    0.00 |
|                                                     |76 VENTED EXCESS MASS  (LBM)    0.00 81 ACCUM GAS SIDE MASS   (LBM)    0.00 |
|                                                     |77 FINAL TANK TEMP       (F)    0.00                                        |
|_____________________________________________________|_____________________________________________________________________________|
```

Figure 20. Typical component main edit screen benchmark area.

3.3 Data Base Line Editor

In addition to the screen editor described in section 3.2, the user may also change component data with the line editor. Either of the editing commands, `E` or `ED`, may be used. A `TYPE` command is also available that enables the user to review a list of current component parameters. These commands are described in the following sections.

3.3.1 E

The `E` (edit with cursor) command allows the user to change data in line edit mode for a particular component in the system. This mode of editing is the default mode when entering a CASE/A model (`TERM;1`). After the `E` command is entered, the crosshairs (cursor) are activated, and the user designates which component is to be edited by placing the cursor on that component and typing any character. The current component data base input parameters are displayed, 19 items per screen, and the user designates changes by an "item number; new value" format. The user may page down in the display by entering "`1`." The edit session can be terminated at any time by entering a "`0`." The updated contents are then displayed.

Format: `E`

Notes: Functionally equivalent to the `ED` command.

3.3.2 ED

The `ED` (edit without cursor) command allows the user to change data in line edit mode for a particular component in the system. The user may be on a graphics or text terminal when issuing this command. This command has the same options and syntax of the `E` command, but the user must indicate the type of component to be edited. This command can be used in the batch mode if required. If more than one component type is present in a case, then CASE/A prompts the user with a numbered list of the component names of that particular equipment type. The user then enters the number of the component that is desired for editing.

Format: `ED;EQTYPE` (option 1)
`ED;EQNAME` (option 2)

Examples: `ED;CHX` (prompts the user with a list of all CHX components in the case if more than one exists).

`ED;LABHX` (edits directly the CHX component name LABHX).

Notes: `EQTYPE` is the mnemonic which is associated with the equipment type (see table 10.0-1). EQNAME is the name given to the component by the user.

An example of the component line editor is shown in figure 21. A list of component parameters is written to the screen 19 lines at a time. The user can then change desired items with an “item number; new value” format. A “1” may be entered to page down the display list. When changes are complete, the user may enter “0” to exit and save the changes. There is no option available in line edit mode to exit without saving changes made in that session.

3.3.3 FASTED

The FASTED (fast edit) command allows the user to change, with a single command, input parameters for all components of a particular type (i.e., CP, PUMP, CHX, etc.) in a case. The user may be on a graphics or text terminal when issuing this command. Changes are specified in an “item #;new value” format and are terminated by entering a “0.”

Format:
```
FASTED;EQTYPE
item #;new value
item #;new value
.
.
.
0
```

3.3.4 TYPE

The TYPE command allows the user to view the component data base for a particular component in the system. The user specifies the type of component to be viewed via the command argument. If multiple components of this type exist in the subsystem, the user is asked to designate the component of interest (a program prompt is provided with a list of choices). The current contents of the component data base are displayed.

Format: `TYPE;EQTYPE`

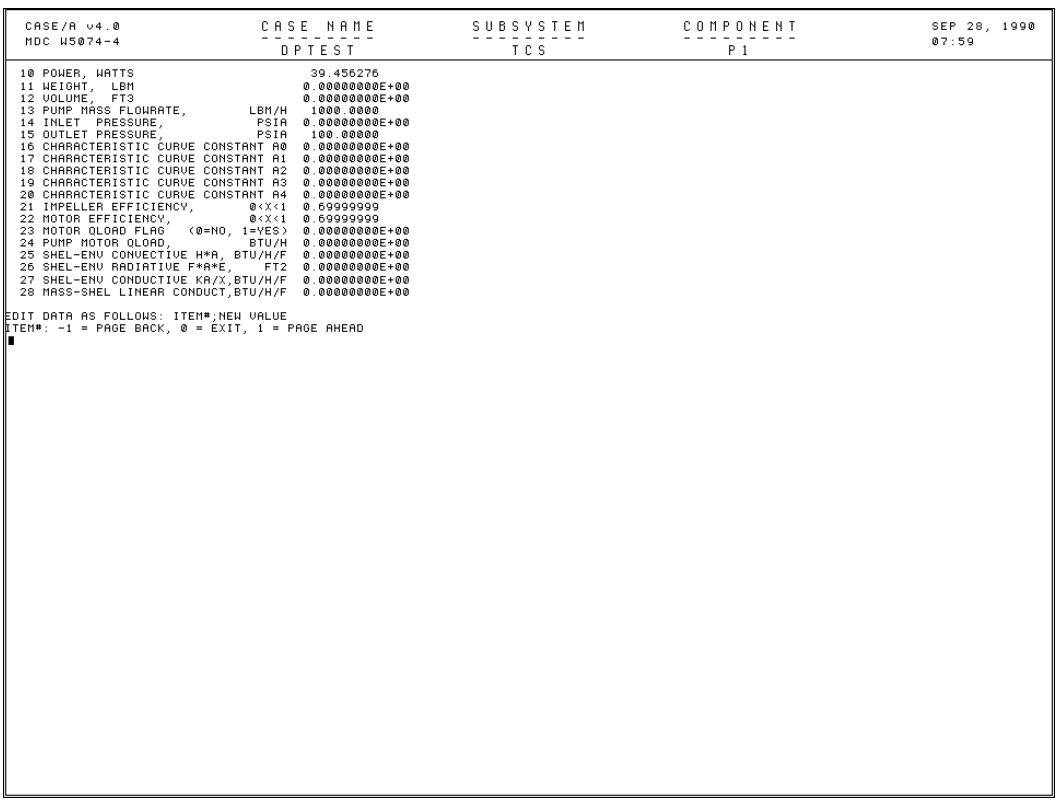


Figure 21. Example of component line editor.

4.0 SIMULATION CONTROL AND EXECUTION

Following the definition of the model components and interconnects, the user may request a simulation of the model over a user-specified time period. The CASE/A program then performs an iterative quasi-steady state solution at each specified time interval until the termination time is reached. The fluid flow rates and compositions remain at constant values over each time interval. The energy balance calculations may be transient or steady-state as dictated by the component type and user input performance parameters.

The solution algorithm operates on an iterative component level basis to achieve convergence of the fluid mass flows and compositions, temperatures, and pressures of all component fluid streams at each time interval. The user may exclude either the pressure or temperature convergence checks by utilizing an optional argument to the solution command. The simulation progress is indicated by status messages written to the display device at the completion of each time interval. The iteration process is described in detail in section 8.1.

4.1 Simulation Control Constants

Control of a CASE/A simulation is accomplished by the interactive specification of control constants contained in the CONTROL data base file record for each model. This is accomplished with the ED command as follows (see section 3.2 for a discussion of the editor):

```
ED;CONTROL
```

The control constants are discussed in the following paragraphs in the order shown in figure 22. The time unit flag can be set to 1 for hours, 2 for minutes, or 3 for seconds. The simulation start time and stop time are input in hours, minutes, or seconds and determine the overall time period to be simulated. The simulation time step is also input in hours, minutes, or seconds and specifies the incremental time period at which the solution system progresses through the simulation. The simulation output interval specifies the incremental time period at which the output summaries are written to the CASENAME.LPP file (see section 5.3). This value should be entered in multiples of the simulation time step. For instance, if the time step is 1 hour but output is desired only for every other hour, the output interval would be set to 2 hours. The plot time units are the units of the time axis on the plots and can be set to hours, minutes, or seconds.

The solution convergence criterion and maximum iterations control the iterative process carried out at each time step during the simulation. The solution iteration process is terminated when the convergence criterion is satisfied or when the maximum number of iterations has been exceeded. The convergence criterion specifies the maximum allowable relative change of each principle parameter on two successive iterations. The principle parameters, which are checked in the CASE/A system, are the component fluid stream mass flow rates, compositions, temperatures, and pressures. See sections 8.1 and 9.3 for further discussion of the convergence process.

The number of constituents to be tracked in the simulation is the next control constant in the edit screen. It is discussed in section 4.2 in detail.

The pressure solution flag indicates the method to be used in the calculation of the component stream pressures: matrix reduction or iterative feedback. This flag should normally be set to 1 to indicate that the matrix reduction method is to be used. A flag value of 0 indicates that the iterative feedback method is to be used. The two methods are discussed in detail in section 8.2. PLEASE NOTE: If the user finds that the model is not converging for cases with a large number of components, please try the "Feedback" solution method (specified in the CONTROL file). In the past, problems with convergence have been noted where the "Matrix" solution method was used for large cases.

The initial pressures of all nonboundary streams at the beginning of the simulation may be set by the user. The default value is 14.7 lb/in^2 absolute.

The flow solution damping factor is used for specialized hydraulic calculations. It has been disabled for version 4.1 of the CASE/A program.

The next five locations of the CONTROL edit screen are used to override the internally calculated thermodynamic properties of any single chemical constituent tracked in the simulation. This is accomplished as follows:

```
CASE NAME: CONTROL            CONTROL EDIT SCREEN           LAST UPDATED: 890410
                                                          RECORD NUMBER: 000001
| TIME UNIT FLAG (1=Hr,2=Min,3=Sec)    1 | PRESS SOLTN,0=Fdbk 1=Mtrx        1. |
| START TIME (Hr, Min, or Sec)     0.000 | INITIAL PRESSURES,   Psia    14.700 |
| STOP TIME (Hr, Min, or Sec)     24.000 | FLOW DMP FAC:M=G*dP,0<x<1 0.000E+00 |
| TIME STEP (Hr,Min,or Sec)      1.00000 | SPECIFIED PROPS CNSTTNT #        0. |
| OUTPUT INTRVAL(H,M, or S)      1.00000 | SPECIFIED CP,    Btu/Lb/F 0.000E+00 |
| PLOT TIME UNITS (1=Hr,2=Min,3=Sec)  1. | SPECIFIED DENSITY, Lb/Ft3 0.000E+00 |
| CONVERGENCE CRITERION, 0<x<1 0.100E-04 | SPECIFIED VISC,   Lb/Ft/H 0.000E+00 |
| MAX SOLUTION ITERATIONS             75 | SPECIFIED CV,    Btu/Lb/F 0.000E+00 |
| NUMBER OF CONSTITUENTS, x<50         9 |                                     |
```

Figure 22. CONTROL default edit screen.

1. Set the specified properties constituent number to the relative system number of the constituent to be overridden (see section 4.2).

2. Set the indicated property values.

The properties of the selected constituent will be set to the user-defined values with no consideration given to the pressure and temperature of the bulk fluid. The properties of all other constituents are calculated based on the pressure and temperature of the bulk fluid. This option provides a quick way to add support for a user-defined constituent without requiring the modification of the system properties calculation routine. The default setting of 0.0 for the specified properties constituent disables this option.

4.2 Chemical Constituents

The number of chemical constituents tracked in a simulation defaults to a minimum of 9 and may be set to a maximum of 50. The first constituent location represents the total mass flow rate of a component stream and is not available for modification. Also, constituents two through nine may not be modified, since several CASE/A routines assume they remain in the default locations shown in figure 23. The user may set the alphanumeric labels of each of the constituents (up to 12 characters long) by editing the LABELS data base record for the active model as follows:

```
ED;LABELS
```

These labels are displayed on the edit screens and in the tabular output file. If the user wishes to track custom constituents in addition to the nine default ones, one mandatory and two optional tasks must be performed:

1. Mandatory: Set the number of constituents to be tracked in the CONTROL data file.

2. Optional: Modify the constituent labels in the LABELS data file.

3. Optional: Modify the FORTRAN logic of the PROPS routine to calculate the thermodynamic properties of the custom constituents.

If the user does not provide properties calculation logic for user-defined constituents in locations 10 to 50, their properties are set to those of liquid water. This is done to at least provide nonzero values for their properties to avoid any arithmetic divide by zero errors. This scheme should prove adequate for low concentration constituents that do not significantly contribute to the bulk properties of a fluid. Custom properties calculation logic may be incorporated by including a modified copy of the PROPS routine in an OPS file to be linked with the CASE/A program. The linkage process is discussed in detail in section 7.4. The user should also refer to section 8.6 for a discussion of the properties calculation methods.

```
CASE NAME: LABELS            LABELS EDIT SCREEN          LAST UPDATED: 890215
SUBSYSTEM: DEFAULT          COMPONENT: VALUES           RECORD NUMBER: 000001
```

CONSTITNT	LABEL	CONSTITNT	LABEL	CONSTITNT	LABEL
CONST 2	OXYGEN	CONST 19	CONSTIT 19	CONST 36	CONSTIT 36
CONST 3	NITROGEN	CONST 20	CONSTIT 20	CONST 37	CONSTIT 37
CONST 4	CARBON DIOX	CONST 21	CONSTIT 21	CONST 38	CONSTIT 38
CONST 5	HYDROGEN	CONST 22	CONSTIT 22	CONST 39	CONSTIT 39
CONST 6	CARBON (S)	CONST 23	CONSTIT 23	CONST 40	CONSTIT 40
CONST 7	METHANE	CONST 24	CONSTIT 24	CONST 41	CONSTIT 41
CONST 8	WATER	CONST 25	CONSTIT 25	CONST 42	CONSTIT 42
CONST 9	FREON-11	CONST 26	CONSTIT 26	CONST 43	CONSTIT 43
CONST 10	CONSTIT 10	CONST 27	CONSTIT 27	CONST 44	CONSTIT 44
CONST 11	CONSTIT 11	CONST 28	CONSTIT 28	CONST 45	CONSTIT 45
CONST 12	CONSTIT 12	CONST 29	CONSTIT 29	CONST 46	CONSTIT 46
CONST 13	CONSTIT 13	CONST 30	CONSTIT 30	CONST 47	CONSTIT 47
CONST 14	CONSTIT 14	CONST 31	CONSTIT 31	CONST 48	CONSTIT 48
CONST 15	CONSTIT 15	CONST 32	CONSTIT 32	CONST 49	CONSTIT 49
CONST 16	CONSTIT 16	CONST 33	CONSTIT 33	CONST 50	CONSTIT 50
CONST 17	CONSTIT 17	CONST 34	CONSTIT 34		
CONST 18	CONSTIT 18	CONST 35	CONSTIT 35		

Figure 23. LABELS default edit screen.

4.3 Plot Data Setup

The IPU allows the user to create a list of simulation parameters (temperatures, pressures, flow rates, etc.) to be saved for later analysis. This parameter list, called a plotset, must be defined prior to execution of the `SOLVE` command. This section outlines the procedures necessary for creating plotsets, as well as how to activate and deactivate existing plotsets. It should be noted that plotsets can be created only while operating on a VT100 compatible terminal. This restriction results from the inherent design of the PLOT data base, which requires the use of a full screen editor. Existing plotsets, however, may be activated or deactivated from a graphics terminal in the line edit mode, if necessary. The PLOT data base is accessed in the same manner as other data bases by using the ED command:

```
ED;PLOT
```

Upon execution of this command, the primary edit screen (full screen mode only) of the PLOT data base will appear as shown in figure 24. Each case may support up to 20 different plotsets. A description of each column follows:

The user-specified plotset name (eight characters or less) is entered in the first column.

The second column, labeled “OPS #”, indicates at which phase of the simulation solution the data are recorded (see section 7.1). The OPS numbers are shown in table 5.

For example, setting the OPS number to 4 would cause the plotset data to be recorded at the end of every timestep.

The third column (SAMPLE) is the sample number. If SAMPLE is set to 1, every point is used. If SAMPLE is set to 2, every other point is used. SAMPLE must be set to at least 1.

The fourth column, PLOT, is the activation code of the plotset: 1 is active, 0 is inactive. If the activation code is set to 0, no data are recorded for the plotset. The user may reactivate the plotset for subsequent simulations by placing a “1” in this column.

The fifth column enables the AUTOPLOT option. If set to 1, the plot file for that plotset will be generated. Subsequently, the AUTOPLOT function can be run to generate all plots for the set (refer to section 5.7).

```
CASE NAME: TEST2B                    PLOT EDIT SCREEN          LAST UPDATED: 890727
                                                            RECORD NUMBER: 000003
|--PLOTSET NAME--OPS #--SAMPLE--PLOT--AUTOPLOT--ARCHIVE--# OF REPETITIONS--^L TO OPEN--|
|  1. BEDPLOT      4      001     1       1        1             01          00039     |
|  2. CABPLOT      4      001     1       0        1             02          00101     |
|  3.              0      000     0       0        0             00          00000     |
|  4.              0      000     0       0        0             00          00000     |
|  5.              0      000     0       0        0             00          00000     |
|  6.              0      000     0       0        0             00          00000     |
|  7.              0      000     0       0        0             00          00000     |
|  8.              0      000     0       0        0             00          00000     |
|  9.              0      000     0       0        0             00          00000     |
| 10.              0      000     0       0        0             00          00000     |
| 11.              0      000     0       0        0             00          00000     |
| 12.              0      000     0       0        0             00          00000     |
| 13.              0      000     0       0        0             00          00000     |
| 14.              0      000     0       0        0             00          00000     |
| 15.              0      000     0       0        0             00          00000     |
| 16.              0      000     0       0        0             00          00000     |
| 17.              0      000     0       0        0             00          00000     |
| 18.              0      000     0       0        0             00          00000     |
| 19.              0      000     0       0        0             00          00000     |
| 20.              0      000     0       0        0             00          00000     |
|_____________________________________________________________________________________|
```

Figure 24. Example plot edit screen.

Table 5. Plotset OPS activation codes.

OPS #	Simulation Phase
1	Simulation initiation
2	Beginning of each timestep
3	Each system iteration
4	End of each timestep
5	Every output interval
6	End of simulation

The sixth column enables the ARCHIVE option. Set this column to 1 to enable the Plotset to be archived. Subsequently, during the ARCHIVE operation, that plotset description can be archived in an ASCII file (refer to sectons 6.7 and 6.8).

The seventh column, NUMBER OF REPETITIONS, is used in the CANNED PLOTSET procedure and must be set to at least 1 at all times. Please refer to section 4.4 Canned Plotset Setup for further explanation.

The creation and/or modification of a plotset is performed in the plotset definition screen. This screen is accessed by positioning the cursor in the last column (labeled ^L to open) and entering the key combination <CONTROL-L>. An example plotset definition screen is shown in figure 25.

The IPU has access to pertinent simulation data such as time, temperatures, pressures, fluid compositions, and component performance parameters. These data are stored in various system data arrays as described in section 7.2. The specific array name is used to indicate the type of parameter to be saved in the plotset. A summary of the system data arrays accessible to the IPU is shown in table 6. The necessary qualifiers are indicated with a check mark. The last column in the plotset definition screen allows the entry of an alphanumeric description (32 character maximum) of the data value.

```
PLOTSET NAME:  BEDPLOT         PLOTSET DEFINITION

 CODE   COMPONENT NAME     LOCATION  STREAM#                  TITLE
 ----   --------------     --------  --------  --------------------------------
 TIME                        0000      0       TIME, HRS
 PRO        BED1             0002      2       OUTLET TEMP BED1, F
 D          BED1             0006      0       OUTLET DEWPOINT TEMP, F
```

Figure 25. Example plotset definition screen.

Table 6. Plotset definition data summary.

Code	Comp Name	Location	Stream No.	Contents
TIME	-	-	-	Simulation time, Hr
PRO	√	√	√	Pressure, temp, etc.
C	√	√	√	Flow rate, compositions
D	√	√	-	Component internal data*
CON	√	√	-	Component performance
USER	-	√	-	User defined constants
NULL	-	-	-	Unused**

* See component descriptions or actual code for specific locations.

** An unused data base location is necessary for certain analysis commands available in the IPU as described in section 5.4.

In the example screen shown in figure 25, a dataset will be created with item 1 as simulation time and a title of “TIME, HRS.” Note that the convention is to place the independent variable (TIME) in the first row and dependent variables following. The second item will be the outlet temperature of a desiccant bed named BED1. The title of the second item is “OUTLET TEMP BED1, F.” This title will be shown in the plot legend in the upper right corner of the resultant plot. The final item is the outlet dewpoint temperature, contained in the internal D array in the component routine, titled “OUTLET DEWPOINT TEMP, F.” The IPU will check that all necessary entries for each active plotset are in the correct range. If an error is discovered, the simulation is terminated and the suspect plotset identified.

A discussion of the keystrokes necessary for moving the cursor in the main edit screen and the secondary edit screen is given in section 3.2. See sample problem 1 in appendix A for a step-by-step discussion of a plotset buildup.

4.4 Canned Plotset Setup

The user has the option to use canned plotsets previously set up to record output parameters from components. Essentially this feature copies a canned plotset to the plot data base as a template for the user to modify to suit the requirements of the model. This feature should save the user time since the plot parameters do not need to be entered from scratch. At this time, the components which have associated canned plotsets include:

Component Name	Canned Plotset Name
CABIN	CABPLOT
AR	ARPLOT
CREW	CREWPLOT
CHX	CHXPLOT

To use a canned plotset simply enter the name of the canned plotset in the first column of the plot edit screen ensuring that there is component output data that are relevant to the canned plotset (i.e., there exists a CABIN component in your model for CABPLOT). Next, set the number of repetitions column to duplicate the parameters in the canned plotset for each component that is to be tracked. For example, if your model contains two cabins, set the number of repetitions to two. This will make a second copy and append it to the first. Standard component names (e.g., LAB, HAB, etc.) are used in CABPLOT and other canned plotsets. When duplicating using the number of repetitions, the order of these standard names is reflected in the duplication (see how component name HAB follows component name LAB in figure 26). This ordering is defined with the AUTOPLOT utility (see the end of this section). Finally, set the other appropriate columns (to activate the plot, etc.) as described in the

previous section. Type Control-L in the last column to open the plot data record and verify what is being recorded. The information recorded can be modified to suit the requirements of the model. When finished verifying type Control-Z to save the record which sets the record number on the plot edit screen (Note: do not use Control-X to exit since this does not save the record number in the plot edit screen).

In the example screen shown in figure 26, a canned plotset data base named CABPLOT will be created with item 1 as simulation time and a title of “TIME, HRS.” The second item will be the outlet temperature of a desiccant bed named LAB. The title of the second item is “LAB TEMPERATURE, DEGF.” This title will be shown in the plot legend in the upper right corner of the resultant plot. The third item is the relative humidity, titled “LAB REL HUMIDITY, %.” There are 10 dependent variables associated with the LAB cabin and 10 similar dependent variables associated with the HAB cabin. Since there were only two repetitions chosen in the number of repetitions column in the plot edit screen (fig. 24), there are only two cabins corresponding in the plotset definition. The plotset definition screen only has 20 lines for components, so the last line has to be viewed on a separate screen, shown in figure 27.

```
PLOTSET NAME:  CABPLOT         PLOTSET DEFINITION

 CODE   COMPONENT NAME     LOCATION  STREAM#                  TITLE
 ----   --------------     --------  --------  --------------------------------
 TIME                        0000      0       TIME, HRS
 CON        LAB              0030      0       LAB TEMPERATURE, DEGF
 CON        LAB              0034      0       LAB REL HUMIDITY, %
 CON        LAB              0034      0       LAB DEWPOINT, DEGF
 CON        LAB              0033      0       LAB PP CO2, MMHG
 CON        LAB              0032      0       LAB PP O2, PSIA
 CON        LAB              0031      0       LAB TOTAL PRESSURE, PSIA
 C          O2LAB            0001      1       LAB O2 FLOW IN, LB/HR
 C          N2LAB            0001      1       LAB N2 FLOW IN, LB/HR
 CON        LAB              0020      0       LAB SENS HEAT LOAD, WATTS
 C          NULL             0001      1       LAB CABIN OPEN LOCATION
 CON        HAB              0030      0       HAB TEMPERATURE, DEGF
 CON        HAB              0034      0       HAB REL HUMIDITY, %
 CON        HAB              0034      0       HAB DEWPOINT, DEGF
 CON        HAB              0033      0       HAB PP CO2, MMHG
 CON        HAB              0032      0       HAB PP O2, PSIA
 CON        HAB              0031      0       HAB TOTAL PRESSURE, PSIA
 C          O2HAB            0001      1       HAB O2 FLOW IN, LB/HR
 C          N2HAB            0001      1       HAB N2 FLOW IN, LB/HR
 CON        HAB              0020      0       HAB SENS HEAT LOAD, WATTS
```

Figure 26. Example plotset definition screen for a canned plotset—first screen.

```
PLOTSET NAME:  CABPLOT         PLOTSET DEFINITION

 CODE   COMPONENT NAME     LOCATION  STREAM#                  TITLE
 ----   --------------     --------  --------  --------------------------------
 C          NULL             0001      1       HAB CABIN OPEN LOCATION
```

Figure 27. Example plotset definition screen for a canned plotset—second screen.

The canned plotset autoplot option allows the user to change the order of the cabin types for certain plotset definitions. Nine cabin types are hard-coded in CASE/A . The user can change the order of the nine cabin types while adding a new canned plotset definition. This is done by using the ED command:

```
ED;AUTOPLOT
```

Upon execution of this command, the edit screen (full screen mode only) of the AUTOPLOT data base will appear as shown in figure 28. The second column titled “CABIN TYPES NEW ORDER” can be filled in to specify the new ordering. Note that all nine slots for the cabin types must be entered. This order will be used in creating the new plotset definition of the canned plotset.

4.5 Simulation Initiation

All connections between components must be completed, and all CONTROL constants must be initialized prior to initiating a simulation. If there are missing connections, the program will terminate the solution and indicate that connections are missing. The request for a solution is made as follows:

```
SOLVE [;optional argument]
```

The optional argument allows the user to disable the convergence checks for either the temperatures or pressures of the component fluid streams. The convergence checks of the mass flows and compositions are always performed. The default solution request (no argument) results in convergence checks on all temperatures and pressures as well as the mass flows of the component streams.

The convergence checks on the pressures may be disabled by use of the optional argument instructing the system to only check the mass flows and temperatures as follows:

```
SOLVE;T
```

The pressures are still calculated as usual but are not checked for convergence. The thermodynamic properties of all fluids will be calculated at atmospheric pressure (14.7 lb/in^2 absolute) inside the PROPS routine.

```
CASE NAME: AUTOPLOT          CABIN TYPES EDIT SCREEN        LAST UPDATED: 890410
                                                          RECORD NUMBER: 000001

|----------CABIN TYPES CURRENT ORDER-----------CABIN TYPES NEW ORDER-----------|
|                                                                               |
|                                                                               |
|                 1. HAB                              1. LAB                    |
|                 2. LAB                              2. HAB                    |
|                 3. N1                               3. ESA                    |
|                 4. N2                               4. JEM                    |
|                 5. ESA                              5. AL                     |
|                 6. JEM                              6. PLG                    |
|                 7. AL                               7. ORB                    |
|                 8. PLG                              8. N1                     |
|                 9. ORB                              9. N2                     |
|                                                                               |
|                                                                               |
|                                                                               |
|                                                                               |
|                                                                               |
|_______________________________________________________________________________|
```

Figure 28. Example autoplot edit screen.

In a similar manner, the convergence checks on the temperatures may be disabled by use of the optional argument instructing the system to only check the mass flows and pressures as follows:

```
SOLVE;P
```

The temperatures are still calculated as usual but are not checked for convergence. The thermodynamic properties of all fluids will be calculated at a temperature of 75 ˚F inside the PROPS routine.

4.6 Simulation Status and Error Messages

Once the simulation has been initiated, the solution progress is displayed at each time step. Also, diagnostic error messages of the component or system routines may be displayed on an iteration or time step basis. The initial stages of the simulation initialization are displayed first, followed by the time step updates as shown in figure 29.

The first parameter displayed in the progress status line is the simulation time in hours. The second parameter, RELX, is the user-defined value of the relative convergence criterion. The third parameter, RXCC, tracks the highest value of the convergence indicators of the component streams. It indicates the greatest change between the last two iterations of all mass flows and compositions, temperatures, and pressures. The RXCC value will be less than RELX if the system reached convergence for the time step. The final parameter, LOOPS, displays the number of system iterations required during the time step to reach convergence.

If the system does not meet the convergence criterion during a time step, a list of the components that did not converge is written to the display device. The listing will differentiate between the mass, temperature, and pressure convergence indicators of the component streams. The convergence indicator values should be examined to see if a modeling problem is occurring, or if an unrealistic combination of convergence criterion and maximum loop count has been specified (see section 9.3).

The individual component or system routines may write error or warning messages to the display device on an intermittent basis. These messages may consist of warnings such as improper performance data input, tanks being completely emptied, or internal component calculations failing to converge. The user may interrupt the simulation to correct the indicated problems by typing a <CONTROL-Y> key combination. This will interrupt the CASE/A program execution and return control to the VAX operating system level. This type of interruption will result in the loss of the simulation benchmark data tracked in the component data base files. The output file data (plotting data, CASENAME.LPP output file, and custom output) will be terminated at the interruption point. The CASE/A program may then be restarted to correct any problems.

```
SOLVE
READING SIMULATION CONTROL DATA...
CHECKING FOR ACTIVE PLOTSETS...
READING COMPONENT DATA BASE PARAMETERS...
CONSTRUCTING PSEUDO-COMPUTE SEQUENCE & FLOW MAP...
PERFORMING COMPONENT INITIALIZATION OPERATIONS...
INITIATING SIMULATION SOLUTION...

**** TIME= 0.0000E+00  RELX=  0.2000E-02  RXCC=0.0000E+00  LOOPS =   0
**** TIME= 0.8333E-02  RELX=  0.2000E-02  RXCC=0.9787E-04  LOOPS =  12
**** TIME= 0.1667E-01  RELX=  0.2000E-02  RXCC=0.9443E-04  LOOPS =   8
     .                 .                  .                 .
     .                 .                  .                 .
     .                 .                  .                 .
**** TIME= 0.1200E+02  RELX=  0.2000E-02  RXCC=0.1463E-02  LOOPS =   2
**** SIMULATION COMPLETED: PERFORMING WRAP-UP...
```

Figure 29. Typical simulation progress display.

5.0 MODEL OUTPUT MANAGEMENT

The following sections discuss the various forms of output supported in the CASE/A modeling package. Methods of obtaining hard copies of subsystem schematics as well as printouts of all component data base parameters are presented. The tabular summary file automatically generated during the solution process is discussed. This file includes valuable information concerning stream conditions and benchmark data at each user-specified output interval. Explanations of the commands necessary to operate the IPU are presented. The IPU allows the user to create a list of parameters (temperatures, pressures, flow rates, etc.) to be saved during a simulation for later analysis. Additionally, methods of customizing output for specialized applications are illustrated. Finally, the connection and flow maps, which are automatically created by CASE/A, are explained in detail to assist the user in debugging a newly created model.

5.1 Schematic Output

There are several options available to the user regarding schematic output. There are commands that refresh the terminal screen to reflect subsystem change, as well as to generate hardcopy graphics on an HP Laserjet printer. Additionally, the user may display stream or component characteristics on the schematic at the appropriate location. Complete descriptions of these commands are provided in the following sections.

5.1.1 LAB

This command allows the user to display stream or component characteristics on the subsystem schematic after a system solution is complete. Three arguments are required to execute this command, as shown below. NC represents the desired constituent, property, or data base parameter as defined in section 7.2. LABEL is the alphanumeric title to be displayed and is limited to eight characters. INA represents the array being referenced: 1 – constituent (C), 2 – properties (PRO), and 3 – data base (CON). After executing the command, the crosshairs will appear allowing the user to specify the desired stream or component to which the label applies by placing the crosshairs in the “hit” box of the stream, or the component for data base values, and typing any character. The crosshairs reappear to allow the user to select the location for the label by typing any character with the crosshairs at the desired location.

Format: `LAB;NC;LABEL;INA`

Example: `LAB;2;TEMP;2` – Places the temperature value of the selected stream with a label “TEMP” at the location indicated by the crosshairs.

5.1.2 MVLAB

This command allows the user to move an existing label to relieve congestion or otherwise improve the subsystem appearance. Upon execution of this command, the crosshairs will appear to allow the user to specify the new location by typing any character with the crosshairs in the new location. The label will appear in the new location, and the old label will be removed upon execution of the next `RD+` (RD “plus”).

Format: `MVLAB`

5.1.3 DELLAB

This command is used to delete an existing label. The crosshairs are activated and the user designates the label to be deleted by placing the crosshairs on that label and typing any character. The label will disappear on the next `RD+`.

Format: `DELLAB`

5.1.4 RD and RD+

The `RD` (redraw) command is used to redraw the current subsystem to clear any unwanted messages from the screen and/or to show the effect of recent component moves, deletions, rotations, etc. `RD+` (RD "plus") is used in conjunction with the `LAB` command so that any generated labels are included on the schematic. Component stream connection “hit” boxes are omitted with the `RD+` command.

Format: `RD or RD+`

5.2 Component Data Base Output

The `PRINTSS` command generates a printout of all data base parameters in a given subsystem. This provides an efficient method of tracking all component data base values for an individual simulation. These printouts may assist the user when searching for any incorrect data base entries which may occur during the assembly of a new case. A complete description of the `PRINTSS` command is provided below.

PRINTSS

This command allows the user to obtain a printout of the current data base parameters for every component in a subsystem. Upon execution of the `PRINTSS` command, the component parameters are written to a file that is submitted to the system line printer spooler. This file is automatically deleted after printing.

Format: `PRINTSS;SSNAME`

Notes: The `PRINTSS` command uses FORTRAN logical unit 18 for creation of the output file.

5.3 Simulation Tabular Summary File

The CASE/A solution routine automatically provides a detailed report of the stream conditions and benchmark values for each component at each simulation output interval (see section 4.1). This information is stored in a file named "CASENAME.LPP" for subsequent viewing by the user via the VAX editor or for transfer to a VAX system line printer for hard copy. The data in the .LPP file are formatted to correlate to the stream designations found in the component descriptions listed in section 10.0, and are sorted by subsystem. Additional information provided in the CASENAME.LPP file includes a summary of the mass status of each STORE unit in the case, a summary of system consumables for both the atmosphere revitalization (AR) system and the water recovery and management (WRM) system, and a power/weight/volume summary of the equipment in the case, as described below. A portion of an .LPP file is presented in figure 30.

The tank summary identifies every STORE device in the case under consideration and prints the initial tank mass, final tank mass, and the change in mass for each tank.

The AR system summary provides pertinent data concerning oxygen consumption, hydrogen consumption, nitrogen consumption, and carbon dioxide production. Also, data are listed for the oxygen generation, carbon dioxide removal, and carbon dioxide reduction subsystems. Water, hydrogen, oxygen, and carbon dioxide balances for the AR system are calculated and displayed as well. It should be noted that the oxygen generation subsystem can be directed to replenish metabolic oxygen consumption only (with other oxygen demands being supplied by oxygen storage tanks), which will lower the water consumption of the oxygen generation subsystem.

The WRM summary displays water requirements (supply needs) and loads (for recovery processing or storage). If the case under consideration contains a PREWAST device, then the WRM summary will provide the output stream quantity of the PREWAST as a load entitled "URINAL MIX." The WASH load is equal to the WASH requirement less latent losses to the cabin.

The power, weight, and volume summary is printed at the end of the .LPP file. Each component data base contains a location for the user to input the component power, weight, and volume. The routine will add these user-input values for each subsystem (subsystem totals), for all equipment assigned to a given cabin (cabin totals), and a grand total for all equipment in a given case (case totals).

The CASENAME.LPP file may be accessed directly from the DCL level by using the VAX/VMS text editor. To access the file from the CASE/A environment, the `EDT` command is used as described below.

EDT

The `EDT` command invokes the standard VAX/VMS text editor. This will allow the user to examine the output file CASENAME.LPP. All standard VAX/VMS EDT commands are available, including exiting to a new file name. Upon executing `EXIT` or `QUIT`, control will be returned to CASE/A.

Format: `EDT;filename.ext`

```
TIME =      0.00 HRS ( 0.0  DAYS)
STEP =      0.00
RELXX= 0.0000100
RELCC= 0.0000000
LOOPS=         0

.....................................................................
....................     SUBSYSTEM:  WCM           ....................
.....................................................................
=====================================================================
CASE: WCM        COMPONENT: WTRTNK     TYPE: STORE
RELATIVE NUM:      1                 SUBSYS: WCM

                       ACCUM
CONSTITUENTS
 TOTAL MASS     LBM       119.00
  WATER        FRACT   1.0000000
FLUID PROPERTIES
  PRESSURE               15.0000
  TEMPERATURE           145.0000
  SPEC HT, CP             1.0000
  DENSITY                61.2950
  VISCOSITY               1.0900
  SPEC HT, CV             1.0000
RELAXATION VALUES
  MASS                 0.0000000
  TEMPERATURE          0.0000000
  PRESSURE             0.0000000
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
TYPE:  STORE

REL NUM:     1

  1  CASENAME                        WCM
  2  SUBSYSTEM NAME                  WCM
  3  COMPONENT NAME                  WTRTNK
  4  NO. OF INPUTS                    63.000000
  5  NO. OF OUTPUTS                   12.000000
  6  NO. OF BENCHMARKS                9.0000000
  7  CDEL RXCC                       0.00000000E+00
  8  TDEL RXCC                       0.16885022E-05
  9  PDEL RXCC                       0.10101017E-01
 10  POWER,WATTS                     0.00000000E+00
 11  WEIGHT, LBM                     0.00000000E+00
 12  VOLUME, FT3                     0.00000000E+00
 13  MAX STORAGE VOLUME, FT3          2.0000000
  .
  .
  .
 94  MAX CONTENTS TEMP, F             145.00000
 95  MIN CONTENTS TEMP, F             144.59030
 96  NOM CONTENTS TEMP, F             144.80740
#####################################################################
TANK NAME:  WTRTNK

INITIAL MASS:    119.00  LBM
FINAL MASS:       11.96  LBM
NET CHANGE:     -107.04  LBM

---------------------------------------------------------------------
                     POWER-WEIGHT-VOLUME SUMMARY
---------------------------------------------------------------------
TOTALS FOR UNASSIGNED SUBSYSTEM: WCM
              POWER  =      0.00  WATTS
              WEIGHT =      0.00  LBM
              VOLUME =      0.00  FT3
TOTALS FOR CASE: WCM
              POWER  =      0.00  WATTS
              WEIGHT =      0.00  LBM
              VOLUME =      0.00  FT3
```

Figure 30. An example CASENAME.LPP file.

Notes: Any text file in the current directory is available to the user. If the filename argument is omitted, the CASENAME.LPP file is assumed.

5.4 Integrated Plot Utility

The IPU allows the user to create a list of parameters (temperature, pressure, flow rate, etc.) to be saved during a simulation for later analysis. The procedure for selecting these values and structuring the associated plot data base is thoroughly described in section 4.3. Note that the parameter list must be defined prior to executing the `SOLVE` command in order for these parameters to be saved. After the simulation is complete, the IPU may be activated by entering `IPU` at the command prompt. The user will then be requested to "ENTER DATA BASE NAME." This data base name corresponds to a plotset name defined as discussed in section 4.3. If the `IPU` command was issued inadvertently, simply hit `<RETURN>` to continue. Otherwise, entering a valid data base name will activate the IPU. A complete list of available commands may be viewed by executing `HELP` at the command prompt to initiate the on-line help utility. Information on a specific command may be received by entering `HELP;command`. The following paragraphs provide explanations of the fundamental commands in the IPU. See sample problem 1 in appendix A for additional information concerning implementation of the IPU.

5.4.1 FULL

The `FULL` command loads all active records into the active set. This command is generally issued prior to performing conditional tests on the data base for generation of a specified subset. The user is advised how many records were loaded into the active set upon execution of `FULL`. Subsequent conditional tests operate on the active set progressively, eliminating unwanted records to result in the desired subset for editing or report operation.

Format: `FULL`

5.4.2 RECORDS

This command provides a list of the record numbers in the active set. This command is usually used for diagnostic purposes.

Format: `RECORDS`

5.4.3 TERM

The `TERM` command tells the system the type of terminal to permit the proper interpretation of editing key inputs and graphics output. A summary of terminal types currently supported follows:

`TERM;1`	Tektronix 4014
`TERM;2`	VT-100, and VT-220
`TERM;3`	DUAL VT100/4014

Format: `TERM;i`

5.4.4 FLAGS

The `FLAGS` command provides a listing of the current state of the operation control flags. The system default settings are shown below. The commands to set each flag "on" or "off" are provided in the last two columns.

SYSTEM FLAG SUMMARY		on	off
1 LOGO	OFF	LOGO	NOLOGO
2 INVERSE FLAG	OFF	INVERSE	NORMAL
3 COMMAND ECHO	OFF	ECHO	NOECHO
4 AUTO HARD COPY	OFF	COPY	NOCOPY
5 PLOT TIME	OFF	TIME	NOTIME
6 DUAL Y AXES	OFF	DUAL	NODUAL
7 VECTOR GRAPHICS	OFF	VECT	NOVECT
8 X AXIS LOG SCALE	OFF	LOGX	NOLOGX
9 Y AXIS LOG SCALE	OFF	LOGY	NOLOGY
10 OVER-WRITE PROTECT	OFF	REUSE	NOREUSE
11 BOLD PLOT	OFF	BOLD	NOBOLD
12 PAUSE FOR COPY	OFF	PAUSE	NOPAUSE
13 NASA LOGO	OFF	NASA	NONASA
14 COLOR FLAG	OFF	COLOR	NOCOLOR
15 TASK COMPLETION EST.	OFF		
16 PLOT LEGEND	ON	LEGEND	NOLEGEND

Format: `FLAGS`

5.4.5 PLOT

The `PLOT` commands generate graphical representation of the data base information in the form of an x-y plot. Four commands exist, differing only in the amount of user-specified title information allowed. The `PLOT` commands are:

PLOT0;N_1;N_2;...N_n	No plot title, no y axis label
PLOT1;N_1;N_2;...N_n y label	Y axis label provided on separate line
PLOT2;N_1;N_2;...N_n y label title	Y axis label and title provided on separate lines
PLOT3;N_1;N_2;...N_n title	Title provided on separate line

The first argument, N_1, is the item number of the independent variable (usually time) which constitutes the x axis of the plot. The next arguments represent the item numbers of one or more dependent variables that will be plotted as the y variables. If more than one dependent variable is provided, different plotting symbols will be utilized to allow identification of the different variables, with the variables and symbols correlated in the legend. The data are normally plotted with straight lines connecting the data points with a symbol evenly spaced in five or six locations to identify the variable. The dependent variable may also be plotted as discrete symbols at every data point by designating the item as a negative number. A mixture of line and symbol plots is allowed. Following the dependent variable list are the optional minimum and maximum y axis ranges. This will override the internally computed y scale. These minimum and maximum values must be in floating point form to avoid being confused with an integer-dependent variable item number. The user may also provide the minimum and maximum x ranges in a similar manner, but the x range may not be specified without the optional y range. The system offers several FLAGS settings that affect the way the plots are displayed. These are summarized as follows:

LOGO,NOLOGO	Controls the display of a LOGO on the plot
NASA,NONASA	Determines the kind of logo displayed (if any)
TIME,NOTIME	Controls the display of time and date on the plot
DUAL,NODUAL	Dual y axes for Farenheit and Centigrade
LOGX,NOLOGX	x axis is logarithmic scale
LOGY,NOLOGY	y axis is logarithmic scale
SETFIT;order	Data will be plotted as discrete points with a curve fit of specified order placed through points. order=-1 deactivates feature

Format: PLOTx;indep;dep 1 [;dep 2; dep 3;{;y min;y max} {;x min;x max}] where x = 0, 1, 2, or 3

5.4.6 COPY and NOCOPY

The copy command sets an internal flag which controls the automatic hardcopy feature. This feature may not be supported in all installations of CASE/A. The feature is deactivated by the NOCOPY command and the status of the flag may be obtained using the FLAGS command. (CASE/A defaults to NOCOPY). While the feature is turned off, the program will bypass the hardcopy operation following PLOTs and the individual pages of some reports. If the user would prefer a pause after each graphics page for viewing or manual hardcopy, the PAUSE flag may be set using the PAUSE command. This flag is turned off with the NOPAUSE command. The program treats the hardcopy command differently for various device types (TERM;x). For instance, in Tektronix mode (TERM;1), the hardcopy device is activated. DEC VT-100 type devices (TERM;2) have no hardcopy capability so a PAUSE is generated instead.

Format: COPY or NOCOPY

5.4.7 TIME and NOTIME

The TIME command determines whether the time and date will appear on the subsequent plots. The default is the NOTIME state which does not generate time and date output. The status of the TIME flag may be verified using FLAGS.

Format: TIME or NOTIME

5.4.8 LOGO and NOLOGO

The LOGO command sets an internal flag which controls whether or not a logo is displayed on subsequent plots. The function is deactivated by the NOLOGO command. The actual logo displayed depends on the status of the NASA flag. If the NASA logo flag is on, the logo will be “NASA.” If the NASA logo flag is off, there will be no logo at this time. It has the potential of displaying another logo there as there was at one

time. The status of all LOGO related flags may be checked using the FLAGS command.

Format: LOGO or NOLOGO

5.4.9 LOGX and LOGY NOLOGX and NOLOGY

These commands set internal flags to determine whether the x or y axis of subsequent PLOTs will be on a logarithmic scale. The commands may be used in any combination to produce the required results (log-log, semi-log x, semi-log y, etc.). Both flags are off (NOLOGX and NOLOGY) by default. The status of these flags may be checked using the FLAGS command.

Format LOGX or NOLOGX,
LOGY or NOLOGY

5.4.10 BOLD and NOBOLD

The BOLD command results in subsequent plots being displayed with a bold line. This was found to result in better readability particularly when the plot is to be used for presentation viewgraphs. The BOLD feature is on by default. In some circumstances (extremely cyclical data) the heavy line is not desirable because the bold lines obscure needed detail in the plot. The feature may be deactivated by the NOBOLD command in these cases.

Format: BOLD or NOBOLD

5.4.11 DUAL and NODUAL

The DUAL command sets an internal flag which determines whether the subsequent plots will have dual y axes for temperature scales in Fahrenheit and Centigrade. This feature is deactivated by the NODUAL command and the status may be checked by the FLAGS command.

Format: DUAL or NODUAL

5.4.12 PAUSE and NOPAUSE

The PAUSE command governs the action taken at the completion of a PLOT and after each page in schedule reports, etc., when the NOCOPY setting is in effect. The command has no effect on operation when the automatic hardcopy feature is in operation (COPY). When the automatic hardcopy function is disabled using the NOCOPY command, plots will be displayed but the next command will be executed immediately with no pause for viewing the data. This condition is desirable when the final product is being generated by the combination of PLOT and subsequent PLTOVR commands, and the intermediate PLOT product is of no interest. When the PAUSE flag is activated, the system will wait for the user to hit a carriage return before proceeding to the next command. During that time, the user may view the data and make manual hardcopies as required. The PAUSE feature is deactivated by the NOPAUSE command and the status may be checked using FLAGS.

Format: PAUSE or NOPAUSE

5.4.13 NASA and NONASA

The NASA command sets an internal flag which determines whether the subsequent plots will have the NASA logo printed on the plot pages. This feature is deactivated by the NONASA command and the status may be checked by the FLAGS command. If the user desires a NASA logo on his/her plots, he/she must set the NASA flag and also set the LOGO flag.

Format: NASA or NONASA

5.4.14 LEGEND and NOLEGEND

The LEGEND command sets an internal flag which determines whether the subsequent plots will have a plot legend identifying the plot curves. This feature is deactivated by the NOLEGEND command and the status may be checked by the FLAGS command.

Format: LEGEND or NOLEGEND

5.4.15 OPEN

The IPU allows as many as four data bases to be open at any given time. The user may toggle between these data bases using the SETDB command. The data base number parameter is the integer 1, 2, 3, or 4. The original data base activated at IPU start-up is automatically allocated to data base number 1. The data base name is the name of the data base to be OPENed. The user may reallocate a data base number to a different

data base. In this case the first data base is closed before the new one is opened. To access an alternate data base, the user issues the SETDB command. The user may then operate on the alternate in the usual manner with all features of the IPU available. The user may return to the original with a SETDB;1 command.

Format: OPEN;data base number; data base name

5.4.16 SELECT

The SELECT command is used to read a data base directly from a file which was created through user defined code. This file must be stored in binary form in a file named FOR0xx.DAT. The data base is then renamed according to the data definition file which defines the data storage structure. The SELECT command requires the user to specify the record length of the data base in number of words.

Format: SELECT;logical number; length in words

Example:

```
ENTER DATABASE NAME: TEST100
SELECT 99 500
TOTAL RECORDS IN TABLE: 120
```

5.4.17 PLTOVR

The plot-over function allows data from another data base to be overlaid onto an existing plot frame. The typical use is the comparison of transient analytical predictions and test data. A typical command sequence for the PLTOVR function is given in the example at the end of this command description.

The user will generally have two data bases active when performing a PLTOVR. The primary data base will be the data base specified at the time of IPU activation and will be designated as data base No. 1. The user will open a second data base using the OPEN command to make the second data set available as data base No. 2. The IPU command echo and automatic hardcopy should be suppressed using the NOECHO and NOCOPY commands.

Prior to beginning the sequence, the user must jump to data base No. 2 (SETDB;2) and configure the active set to contain the points which will be plotted with the data from data base No. 1. This may involve the FULL, TARGET, SAMFST, RANGE, etc. commands as required. The user will then return to data base No. 1 (SETDB;1) and command a PLOT of the desired information from data base No. 1. Following the creation of the data base No. 1 plot, the user may command the overlay of x_item, y_item data from data base No. 2 using PLTOVR (Note: the user does not leave data base No. 1 to accomplish this). Multiple y_items may be specified, and the display of data can be connected points (positive y_item) or discrete points (negative y_item). The user may generate a hardcopy using the HDCOPY command following the PLTOVR.

Format: PLTOVR;DB number; x_item1;y_item1 [;y_item2;y_item3..]

Example:

ENTER DATABASE NAME $ ANALYSIS	Data base #1 will be ANALYSIS
IPU_ANALYSIS> **FULL**	Prepare to plot all data in ANALYSIS
NUMBER OF RECORDS IN TABLE: 500	System response 500 points found
IPU_ANALYSIS> **OPEN;2;TESTDATA**	Open data base #2 as TESTDATA
IPU_ANALYSIS> **SETDB;2** NUMBER OF RECORDS IN TABLE: 0	Jump to TESTDATA Active set is empty
IPU_ANALYSIS> **FULL**	Find all data in TESTDATA
NUMBER OF RECORDS IN TABLE: 10500	System found 10500 points
IPU_ANALYSIS> **SAMFST;20** NUMBER OF RECORDS IN TABLE: 527	Use only every 20th point 527 points remain
IPU_ANALYSIS> **SETDB;1**	Jump back to ANALYSIS
NUMBER OF RECORDS IN TABLE: 500	ANALYSIS active set is still at 500
IPU_ANALYSIS> **TERM 5**	Designate HP Laserjet
IPU_ANALYSIS> **NOCOPY**	Suppress automatic hardcopy
IPU_ANALYSIS> **PLOT2;1;16**	PLOT ANALYSIS x=item 1, y=item 16
PRESSURE PSIA TRANSIENT PRESSURE PROFILE	y axis label Plot title
[PLOT DISPLAYED]	
IPU_ANALYSIS> **PLTOVR;2;2;-23**	Overlay TESTDATA x=item 2,y=item 23 Note: y item is negative so it will be plotted as discrete points
[OVERLAY DISPLAYED]	
IPU_ANALYSIS> **HDCOPY**	Command a hardcopy of the composite plot

5.4.18 LIST

The LIST function produces a listing of the requested record. The item numbers, item titles, and item field contents are listed in a sequential manner. Text fields which extend beyond 32 characters are truncated. Other ways to examine a record's contents include the PRINT, REPORT, and EDIT functions.

Format: LIST;record number

5.4.19 MODIFY

Allows the user to modify several items in a given record. The user is prompted to enter data in an "ITEM;VALUE" format, one per line. Any number of items can be changed in any order. The operation is terminated by a zero item number. The entire record is listed upon completion.

Format: MODIFY;record number

Example:
```
MODIFY;1001
2;NEW TEXT
16;40.32
0
```

5.4.20 FAST

The FAST command is used to make modifications to certain items of every record in the active set. It operates in an identical manner to the MODIFY command except the scope extends from a single record up to the entire active set.

Format: FAST

Example:
```
FAST
ENTER RECORD CONTENT LINES
FORMAT--ITEM;VALUE
12;NEW TEXT
15;34.90
9;10
0
```

5.4.21 PRINT

The PRINT command generates a headed column printout of the specified items in the active set. The item titles in the heading are taken from the .DDF file for the data base. The number of items is optional but the amount of data to be printed cannot exceed 132 columns. The number of character positions allocated to an item can be set by appending =xx to the item specification in the command. The record number may be printed by specifying a zero item number.

Format:
```
PRINT;item1
[;item2;item3...;itemn]
```

Example: PRINT;0;12;16;2;5=20

In the example, the record number (item 0) will be printed in the first column followed by item 12, item 16, item 2, and item 5. The column width for item 5 will be 20 characters wide. If Item 5 is a 60 character alphanumeric field, for example, the data will be automatically word wrapped to multiple lines as required. The system will generate a pause at each full page of information when operating in interactive mode (TERM;1,2,3, or 6). The PRINT function may be discontinued at the pause by typing an X instead of only a carriage return. When operating in Batch mode, (TERM;4), no pause will occur.

5.4.22 TECPRT

The TECPRT command generates a headed column printout of the specified items in the active set and is tailored to operate on the Tektronix type device. This routine is similar to the PRINT routine which is tailored to VT-100 type devices. The item titles in the heading are taken from the DDF file for the data base. The number of items is optional, but the amount of data to be printed cannot exceed 132 columns. The number of character positions allocated to an item can be set by appending =xx to the item specification in the command. The record number may be printed by specifying a zero item number.

Format:
```
TECPRT;item1[;item2]
title
```

Example:
```
TECPRT;0;12;16;2;5=20
report title
```

In the example, the record number (item 0) will be printed in the first column followed by item 12, item 16, item 2, and item 5. The column width for item 5 will be 20 characters wide. If Item 5 is a 60-character alphanumeric field, for example, the data will be automatically word wrapped to multiple lines as required. The title submitted on a separate line will appear at the top of each page. The system will generate a pause at

each full page of information when the automatic hardcopy is turned off. The routine requires the terminal type set to 1 (`TERM;1`).

5.4.23 OUTPUT

The `OUTPUT` command will redirect all IPU related output to the specified VAX VMS file name. The output can be returned to the user's terminal with an `OUTPUT;TERMINAL` command.

Format: `OUTPUT;file name`

5.4.24 RANGE

The `RANGE` function tests each record in the active set to determine whether the designated item lies within the specified low value-high value range (inclusive). If the test is positive (within range), the record is retained in the resulting active set. If the test is negative (outside range), the record is eliminated from the active set (but not deleted from the data base). The user is advised how many records passed the conditional test upon completion of the function. The `INVERSE` command reverses the function of the range command so that the new active set contains records where the designated item is outside the specified range.

Format: `RANGE;item;low value;high value`

5.4.25 TARGET

The `TARGET` function tests each record in the active set to determine whether the designated item equals the specified value. If the test is positive (equal to value) the record is retained in the resulting active set. If the test is negative (not equal to value) the record is eliminated from the active set (but not deleted from the data base). The user is advised how many records passed the conditional test upon completion of the function. The `INVERSE` command reverses the function of the `TARGET` command so that the new active set contains records where the designated item is not equal to the specified value.

Format: `TARGET;item;value`

5.4.26 SAMFST

The `SAMFST` function will reduce the size of the active set by selecting points on the specified frequency. A `SAMFST;20`, for instance, will select every 20th point in the active set. This is a quick way to reduce the size of large data bases, typical of test data. The routine will always include the first and last point in the active set.

Format: `SAMFST;frequency`

5.4.27 SIGMA

The `SIGMA` function performs a statistical filtering of the data in the active set to remove points which are more than the specified number of standard deviations from the rest of the data points over a particular time step. The routine performs a stepwise scan of the active set in accordance with the specified time step. Time is assumed to be in Item 1 of the data base. For each step a mean and standard deviation are computed for the data points of the specified item. Each step is then rescanned to eliminate data points which are more than the specified number of standard deviations from the mean over the interval. The function is particularly useful in systematically eliminating noise from recorded test data or flight data.

Format: `SIGMA;item;step;standard_deviation`

Example:

```
IPU_DBNAME> FULL                    Start with FULL set
TOTAL RECORDS IN TABLE: 15001       15001 records found

IPU_DBNAME> SIGMA;12;5.0;3.0        Remove records where item
                                    12 is more than

TOTAL RECORDS IN TABLE: 14501       3 sigma out over a 5.0 h
                                    interval
```

5.4.28 TABLE

This function is used to manually create an active set. The user provides the records to be inserted into the table one per line and terminated by a zero entry. For entry of a large number of records it is recommended that the user place the table command and associated record numbers in a standard VAX text file using the VAX system editor of your choice. Access to VAX DCL and the editor is provided by the DCL command without leaving IPU. Performing the TABLE function in this manner will allow the user to repair errors

in the record number entries without retyping the entire string. The text file can be executed as a macro using the RUN command.

Format: TABLE

5.4.29 DTABLE

This command deletes all records in the active set. As with the DELETE function the records are simply flagged as inactive with the information in the records preserved. The data is not destroyed until it is overwritten by new data being entered into the data base. Up to that time the data may be returned to active status with the RESTOR command.

Format: DTABLE

5.4.30 DELETE

This command is used to delete the specified record from the data base. The record is marked for deletion and is inaccessible to subsequent FULL commands. The data is not destroyed until the record is actually overwritten. Up to that point the data may be returned to active status with the RESTOR command. For deletion of many records the user may use the DTABLE command. This command marks all records in the active set for deletion and places the data in inactive status.

Format: DELETE;record

5.4.31 DCALC

This command generates a concise summary of the contents of the CALC registers and the memory locations. The format of the output is as follows:

```
CALC  MEMORY

1     0.000000
2     0.000000
3     0.000000
4     0.000000
5     0.000000
6     0.000000
7     0.000000
8     0.000000
9     0.000000
10    0.000000

CALC  STACKS

1     0.000000
2     0.000000
3     0.000000
4     0.000000
5     0.000000
6     0.000000
7     0.000000
8     0.000000
9     0.000000
10    0.000000
```

Format: DCALC

5.4.32 CLCALC

This command will zero the CALC registers (stacks) and the 10 memory locations. This command is generally issued prior to beginning a new CALC sequence.

Format: CLCALC

5.4.33 CALC

The CALC (calculation) function allows the user to mathematically modify the data base contents in accordance with a user-specified algorithm. The CALC function is designed to emulate a reverse polish notation (RPN) calculator keystroke programming method (Hewlett-Packard calculator). The CALC function operates on each record in the active set. The CALC function has a 10 register stack and 10 memory cells available for use in accomplishing the calculations specified. The stack operates in the following manner:

- A retrieved item or user specified constant is placed into the lowest stack location, pushing all other data in the stack up one location.

- Two number operations (addition, multiplication, etc.) operate on the bottom two locations. Both numbers are destroyed in the process with the result of the operation in the lowest stack location. The data which was in the third location and above fall one location.

- One number operations (sine, log, 1/x, etc.) operate on the lowest location. The operand is destroyed with the result now occupying the lowest location.

- A command to store data takes the information from the lowest location to a place in the specified data base item or storage cell. The stack contents are not affected.

The user must provide the “key-stroke” sequence consisting of operations and arguments from the command set described below. Each operation is submitted on a separate line. The user terminates the input of the calculation sequence with `<Ctrl Z>`. The calculation sequence is then applied to each record in the active set. Long or complex sequences should be executed in the form of a macro using the `RUN` command. The routines `CLCALC` and `DCALC` can be used before and after the `CALC` execution as required. The user should treat a `CALC` sequence as any other program which operates on valuable data base information. Copies of the data base should be made prior to “debugging” the algorithm to allow recovery from a malfunctioning command sequence.

The following provides an overview of the available `CALC` operations. Following the convention frequently used in the Hewlett-Packard manuals, the lowest two stack register locations will be designated as: x for register 1, and y for register 2.

`I;item`	Places the specified item in x, pushes the rest up one register.
`M;location`	Places the contents of the memory location specified into x, pushes the rest up one register. The location argument may be an integer from 1 to 10.
`SI;item`	Places the contents of the x position into the data base at the item specified.
`SM;location`	Places the contents of the x position into the memory at the location specified (integer from 1 to 10).
`GOTO;step`	#Transfers control to the specified step number.
`E;value`	Places the constant "value" into x, pushes the rest of the registers up in the process.
`IFEQ;step`	If register x .EQ. register y go to the specified step.
`IFLT;step`	If register x .LT. register y go to the specified step.
`IFLE;step`	If register x .LE. register y go to the specified step.
`IFGT;step`	If register x .GT. register y go to the specified step.
`IFGE;step`	If register x .GE. register y go to the specified step.
`IFNE;step`	If register x .NE. register y go to the specified step.
`+`	Add x and y, x and y are "destroyed", x = sum.
`-`	Subtract x from y, x and y are "destroyed", x= difference.
`*`	Multiply x and y, x and y are "destroyed", x = product.
`/`	Divide x into y, x and y are "destroyed", x = quotient.
`**`	Raise y to x power, x and y are "destroyed", x = result.
`SIN`	x = SIN(x), x in radians
`COS`	x = COS(x), x in radians
`TAN`	x = TAN(x), x in radians
`ASIN`	x = ARCSIN(x), x in radians
`ACOS`	x = ARCCOS(x), x in radians
`SINH`	x = SINH(x)
`COSH`	x = COSH(x)
`TANH`	x = TANH(x)
`EXP`	x = EXP(x)
`LN`	x = LN(x)
`SQRT`	x = SQRT(x)
`LOG`	x = ALOG10(x)
`1/X`	x = 1/x
`X-Y`	x = y, y = x
`CF`	convert x to °F from °C
`FC`	convert x to °C from °F
`ABS`	X = ABS(x)
`END`	end program

Format: `CALC`

5.4.34 AVERAGE

The `AVERAGE` function generates an integrated average of all items in the data base at the specified time interval. The results are written to the alternate data base number (1, 2, 3, or 4) specified. The alternate data base is opened prior to the execution of the routine. The data base should usually have the same data definition file (.DDF) contents as the original but under a different name. The active set will contain the record numbers of the averaged data set. The user must identify the item location of the time

variable in the record. The results appear as stepwise data on the specified interval.

Each step has both a start time and a stop time written to the alternate data base. This results in the end time of the Nth step coinciding with the starting time of the (N+1)th step. If the time step is greater than the time step of the data in the data base, a trapezoidal rule integration is performed on the data points within the interval. Appropriate interpolation is applied to assure that the first and last points in the time step are properly handled under the condition that only part of a data interval falls within the time step in question. If the time step is smaller than the time step of the data in the data base the averaged data will appear as a stepwise linear interpolation between the points which exist.

Format: `AVERAGE;timestep;`
`alternate DB number;`
`time item`

5.4.35 RUN

The `RUN` command is used to execute a set of previously defined commands as a macro. The macro may include any IPU command, the `EXIT` command, and the `DCL` command followed by its associated VAX/VMS commands. The command set must be placed in a standard VAX text file and may be created using one of the system-resident editors. If desired, the EDT command may be used to create the command set while operating in the CASE/A environment. The `RUN` command will execute all commands in the macro and return control to the user when complete.

Format: `RUN;file name`

5.5 Custom User Output

As the user becomes familiar with the development and implementation of User Operations Logic (described in section 7), the need may arise for customizing output information. An example would be any user-defined variable which is to be stored for later analysis but is not available to the IPU through any of the system data arrays (see section 7.2). In such a case, the user has two distinct options depending on the complexity of the particular situation. The first option, which is the preferred method, is to use the USERCON array available in all OPS blocks. This array provides 100 user-accessible locations and is recognized by the IPU. The user simply stores the parameter to be tracked into any unused USERCON location. This value is accessed through the IPU as described in section 4.3. The second and more complex method is for the user to create a binary data file for storing any desired variables. The user must store these data according to a data definition file (.DDF) which must be created following the guidelines in the CASE/A programmer's manual. After the simulation is complete, the user may manipulate these data from within the IPU after loading the data base with the `SELECT` command. The `SELECT` command and all related IPU commands are discussed in the previous section.

5.6 Schematic Connection and Hydraulic Maps

The solution routine automatically generates output files containing schematic connection and hydraulic flow conductor information to assist the user in debugging new or existing cases. The schematic connection map is created in the user's directory and named CASENAME.CMP. The .CMP file lists all components in the case, sorted by relative equipment numbers, and indicates the type and name of each, as well as the subsystem in which it resides. The next block of information provides a list of all connections in the case. The last block shows the pseudo compute sequence as described in section 7. An example of a .CMP file is shown in figure 31. The hydraulic flow map is also created in the user's directory and is named CASENAME.FMP. This file contains flow conductor information and a breakdown of the flow network into subnetworks. Information regarding flow calculations and pressure drop solution methodologies is provided in section 8.2. An example of an .FMP file is shown in figure 32.

```
*************************************************************************

                      CONNECTION MAP FOR CASE ATC

*************************************************************************
    NUMBER OF COMPONENTS    4     NUMBER OF CONNECTIONS   5
    LENGTH OF PSEUDO    6


                           COMPONENT LISTING

REL                       CENTROID   CONNECTION OF EACH COMPONENT STREAM
IEQ  SUBSYS  ID   NAME    X    Y      1    2    3    4    5    6    7    8
--- -------- -- -------- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----
  1 ATC      27 P-1       281  397    1    3    0    0    0    0    0    0
  2 ATC      33 PAYLOAD   529  543    4    1    0    0    0    0    0    0
  3 ATC      32 LOAD-1    385  260    3    4   14    0    0    0    0    0
  4 ATC      50 C-1       479  387    0    0    0    0    0    0    0    0


                          CONNECTION LISTING

      REL              REL
CONN  IEQ STREAM  TO  IEQ STREAM  NAME OF COMPONENTS
---- ---- ------     ---- ------  ------------------
   1    1    1   ==>    2    2    P-1     : PAYLOAD
   2    4    2   ==>    1   99    C-1     : P-1
   3    1    2   ==>    3    1    P-1     : LOAD-1
   4    3    2   ==>    2    1    LOAD-1  : PAYLOAD
   5    4    1   ==>    3   -2    C-1     : LOAD-1


                        PSEUDO COMPUTE SEQUENCE

SEQUENCE    CONN     REL              REL
 NUMBER    (NCPT)    IEQ STREAM  TO  IEQ STREAM
--------  --------  ---- ------     ---- ------
    1         1       1     1  ==>    2     2
    2         3       1     2  ==>    3     1
    3         4       2     1  ==>    3     2
    4         1       2     2  ==>    1     1
    5         3       3     1  ==>    1     2
    6         4       3     2  ==>    2     1
```

Figure 31. An example .CMP file.

```
*********************************************************************
                        FLOW MAP FOR CASE ATC
*********************************************************************
    NUMBER OF EQUIPMENT ITEMS:       4
    NUMBER OF PRESSURE NODES:        6
    NUMBER OF FLOW CONDUCTORS:       5
    NUMBER OF FLOW SUBNETWORKS:      1
    NUMBER OF CNTRLLR COMPONENTS:    1
    NUMBER OF TIMER COMPONENTS:      0
    NUMBER OF BUBBLE COMPONENTS:     0
    *** NOTE: CNTRLLRs, TIMERs, AND BUBBLEs DO NOT  ***
    *** APPEAR IN THE NODE INFORMATION LIST BELOW   ***
                      PRESSURE NODE INFORMATION
SYSTEM REL                                            CONN
 NODE  IEQ  SUBSYS  COMPONENT TYPE   NAME   STREAM (NCPT)
------ --- -------- -------------- -------- ------ ------
    1    1 ATC       27  PUMP      P-1          1      1
    2    1 ATC       27  PUMP      P-1          2      3
    3    2 ATC       33  TBUS      PAYLOAD      1      4
    4    2 ATC       33  TBUS      PAYLOAD      2      1
    5    3 ATC       32  CP        LOAD-1       1      3
    6    3 ATC       32  CP        LOAD-1       2      4
                      FLOW CONDUCTOR INFORMATION
 SYSTEM   SYSTEM    SYSTEM
CONDUCTOR NODE A TO NODE B  COMPONENT A STREAM TO COMPONENT B STREAM
--------- ------    ------  ----------- ------    ----------- ------
     1        1  ==>    4    P-1           1   ==> PAYLOAD       2
     2        2  ==>    5    P-1           2   ==> LOAD-1        1
     3        3  ==>    4    PAYLOAD       1   ==> PAYLOAD       2
     4        3  ==>    6    PAYLOAD       1   ==> LOAD-1        2
     5        5  ==>    6    LOAD-1        1   ==> LOAD-1        2
                     FLOW SUBNETWORK INFORMATION
==========================================================
NETWORK NUMBER    1 CONTAINS    6 NODES AND    5 CONDUCTORS
  NETWORK    CORRESPONDS    SYSTEM
NODE NUMBER      TO       NODE NUMBER
-----------  -----------  -----------
      1          ==>            1
      2          ==>            4
      3          ==>            3
      4          ==>            6
      5          ==>            5
      6          ==>            2
  NETWORK    CORRESPONDS    SYSTEM
 CONDUCTOR       TO        CONDUCTOR
-----------  -----------  -----------
      1          ==>            1
      2          ==>            3
      3          ==>            4
      4          ==>            5
      5          ==>            2
```

Figure 32. An example .FMP file.

5.7 The Autoplot Process

The autoplot process provides the user with the automated capability to produce a file of all possible plots in a plotset for a solution run (this capability is new in CASE/A verison 5.0). Based on the output parameters selected for a solution run plotset, the independent variable, usually Time, is plotted against each of the dependent variables selected in that plotset. Each plot is written to a file in the user's CASEA directory, whose name is a concatenation of the case name, plotset name and a .PLT extension as follows:

```
CASENAME_PLOTSETNAME.PLT
```

Please follow these steps to obtain plots through the Autoplot process:

(1) Before running the SOLVE process, edit the PLOT data base (using the ED PLOT command) and set the following columns for the plotset you wish to obtain plotsets for.

a) Place a "1" in the PLOT column
b) Place a "1" in the AUTOPLOT column
(Refer to figure 24).

(2) The solution can be run at this point as is normally done.

(3) After the solution run, start the IPU and provide the name of the plotset selected above. Next, enter the command AUTOPLOT. The user has a choice of either batch or interactive submission of the autoplot process. It is recommended to use batch submission when a large number of plots are produced, that is, where over 15 or so dependent variables are in the plotset. The batch submission will run the autoplot process in the background which allows the user to immediately continue other work. After the batch job is submitted to the queue, a message will be displayed showing the entry number of the job that was submitted to the batch queue.

(4) Once the AUTOPLOT process has produced the autoplot file:

CASENAME_PLOTSETNAME.PLT

the user can submit this file to be printed interactively or in batch mode with a command similar to the following:

```
$ PRINT/NOTIFY/QUE=LP_4610_3045_EXC/ -
PARAM=(DATA_TYPE=TEK4014,PAGE_ORIEN=LAND) -
CASENAME_PLOTSET NAME.PLT
```

If the file contains many plots, it is more practical to submit the print job to the batch queue.

6.0 UTILITY COMMANDS

The following sections describe a variety of commands available within the CASE/A environment which are designed to make system use more efficient. These commands are not usually required for normal simulation execution and did not conveniently fall into the command lists provided in sections 2 through 5. It would benefit both new and experienced users, however, to become familiar with these commands.

6.1 Terminal Settings

The CASE/A modeling package is designed to operate on terminals capable of emulating Tektronix graphics terminals and/or DEC VT100 text terminals. The preferred configuration would be an advanced graphics terminal capable of emulating both. The default terminal type is the Tektronix graphics terminal (4014 or compatible). The terminal type may be changed by the user to match a particular system. Additionally, control of baud rate is provided to support the Tektronix 4014-1 storage tube devices. The user may examine the status of a variety of terminal related settings using the FLAGS command. Complete descriptions of these commands are provided in the following sections.

6.1.1 TERM

The `TERM` command defines the type of terminal used in a given CASE/A work session to permit the proper interpretation of editing and output commands. The following is a summary of the terminal types currently supported.

`TERM 1` – Tektronix 4014, 4107
`TERM 2` – VT100, VT200
`TERM 3` – Dual VT100/4014

Format: `TERM;i`

6.1.2 FLAGS

This command displays the status of a variety of various system flags such as `COPY/NOCOPY`, `VECT/NOVECT`, `BOLD/NOBOLD`, etc.

Format: `FLAGS`

6.2 On-Line Help Information

An on-line `HELP` utility is available anytime during a work session at the CASE/A command prompt. A complete description of the `HELP` command is provided below:

HELP

Executing `HELP` will provide a listing of all available CASE/A commands. The listing is divided into three distinct categories: (1) model development and solution commands, (2) data management commands, and (3) engineering analysis commands. To display information regarding a specific command, simply execute `HELP;command`. At the conclusion of any help session, the user is automatically returned to the command prompt.

Format: `HELP[;command]`

6.3 Temporary Exit to VAX Operating System

The user may temporarily exit the CASE/A environment and enter the VAX operating system by executing the `DCL` command as described below:

DCL

The `DCL` command allows the user to enter the VAX DCL (digital command language). The user may then execute any DCL level command with the exception of changing the default directory. Operations may be performed on files outside the current directory, however, by simply using the appropriate disk and directory references in the commands. The user returns to the CASE/A environment by entering LOGOUT or LO at the DCL prompt.

Format: `DCL`

6.4 Temporary Exit to VAX Editor

The user may temporarily exit the CASE/A environment and directly enter the VAX/VMS

text editor by executing the EDT command. A complete description of the EDT command is provided below:

EDT

This command involves the VAX/VMS text editor allowing the user to examine the output file CASENAME.LPP or modify the operations file CASENAME.FOR. Any text file in the current directory is available to the user. All of the VAX/VMS EDT editor commands are usable. When the user "EXIT"s or "QUIT"s the editor, control is returned to CASE/A.

Format: `EDT[;filename.ext]`

Notes: If the argument is omitted, the CASENAME.LPP is entered.

6.5 Subsystem Heat Load Assignment to Cabins/Modules

The CABIN component is used to simulate a cabin, compartment, or otherwise isolated environment by tracking the respirable atmospheric compositions, mass additions/losses, and heat transfer between equipment, other cabins, and the external orbital environment. For the CABIN to recognize the equipment contained in it, the equipment must be assigned to the CABIN using the `ASSIGN` command, as described below. For a complete description of the interactions between cabins and the associated equipment, see section 10.0.

6.5.1 ASSIGN

This command assigns a subsystem to a cabin. This provides the capability to address multiple cabins with different configurations and/or environments. The `ASSIGN` command must be issued before executing `SOLVE`. If the user fails to make a cabin assignment, the cabin air, structure, and mean radiant environment will default to 75 ˚F from a component standpoint. The user should zero out the conductances for the components that are assumed to operate adiabatically. If the interaction with the environment is disabled, then the `ASSIGN` command is not required.

Format: `ASSIGN;SSNAME;CABINNAME`

6.5.2 UNASSIN

This command unassigns a subsystem from a cabin. It is functionally opposite of the `ASSIGN` command.

Format: `UNASSIN;SSNAME`

Subsystem heat load assignments also apply to the MODULE component that was added in CASE/A version 5.0. The MODULE component differs from the CABIN in that three different air volumes are considered in the MODULE, whereas only one is considered in the CABIN. The ASSIGN command is used to assign subsystems to a MODULE in the same way described above for CABINs except that the subsystem is assigned to and tracked within only the "rack" volume of the MODULE.

6.6 Merge Operation Across VAX Hosts

The `MERGE` command is particularly useful when developing large and complicated models. Complete subsystems are constructed and verified as small, manageable cases. The individual subsystems are then merged into the larger model. The smaller case may reside in the same directory, a different directory, or in a directory on a different VAX host than where the larger destination model exists. In addition, CASE/A version 5.0 provides the capability to merge cases from previous versions, thus allowing the user to incorporate subsystems from versions prior to 5.0 into new cases running under version 5.0. The `MERGE` command is described below:

MERGE

This command allows the user to include a complete subsystem (or the entire model) from a different case in the current case. Complete subsystems can be built and verified as separate cases and then the subsystem merged into the larger model. Development time is reduced since the models are developed as smaller, more manageable problems and then assembled into a single model. The routine will check for duplicate component names and solicits a new name from the user if necessary. If the user does not want to change the name, a carriage return is entered and execution continues.

Format: `MERGE;CASENAME;SSNAME`

NOTE: If the case name or subsystem name is in error or omitted, the program will solicit a new input. The routine prompts the user to enter the location of the source CASENAME.MOD file and the source data base files. These locations must be of the following form:

```
Host"username password"::
Device:[Directory]
```

Host and device are necessary only if different from current case.

6.7 Archive Setup Process

CASE/A gives the user the option of archiving any simulation solution through the Archive Process. Each case may support up to 20 different archive sets. Prior to running each case, the user can specify which archive set to associate the case with and after running the model, execute the ARCHIVE command to copy the specified archive sets. This provides the ability to recall the case (using the RETRIEVE command) for comparison against similar solution runs or to vary the model slightly to evaluate variants of the model.

Two screens are associated with the archive setup process which must be filled in before initiating the ARCHIVE command (the ARCHIVE command is discussed in section 6.8). The first is accessed in the same manner as other data bases by using the ED command:

`ED;ARCHIVE`

Upon execution of this command, the primary edit screen (full screen mode only) of the ARCHIVE data base will appear as shown in figure 33. The user-specified archive set name (eight characters or less) is entered in the first column. The second column, labeled "ACTIVE", indicates whether or not the user chooses that archive set for that particular archive process: 1 is active, 0 is inactive. Only one archive set may be set to active for an archive process. If the activation code is set to 0, no data are recorded for the archive set but the user may reactivate the archive set for subsequent archive actions.

The second screen for each archive set is called the archive data base screen. It is accessed from the first screen by positioning the cursor in the last column (labeled ^L TO OPEN) and entering the key combination <CONTROL-L>. The user's comments pertinent to an archive set are stored in the archive data base screen shown in figure 34. This screen provides the space for the user to document each run so that the run can be distinguished from other similar runs. The maximum size for each field follows:

Title	48 characters
Study	56 characters
Purpose	216 characters
Model Description	192 characters
Results	216 characters
Run Specific Param.	196 characters

Because several of these descriptive data narratives take more than one line, the user may continue typing without entering a carriage return. The horizontal arrows may be used to space along the line to the next line.

Both the archive edit screen and the archive data base screen are exited in the same manner as other data base screens: with a ^Z to save the information or with a ^X if the user does not wish to save what was entered.

In addition to the archive data base, the plot data base must also be updated to specify which plotsets are to be archived. This is done by setting a 1 in the archive column of the plot edit screen for each plotset of the case. Please refer to section 4.4, Plot Data Setup, for a view of the screen and more detailed information.

```
CASE NAME: ATC                 ARCHIVE EDIT SCREEN          LAST UPDATED: 890727
                                                          RECORD NUMBER: 000003
|-------ARCHIVE SET NAME-------------ACTIVE-------------^L TO OPEN-------------|
|        1. TEST1                     1                   00033                |
|        2.                           0                   00000                |
|        3.                           0                   00000                |
|        4.                           0                   00000                |
|        5.                           0                   00000                |
|        6.                           0                   00000                |
|        7.                           0                   00000                |
|        8.                           0                   00000                |
|        9.                           0                   00000                |
|       10.                           0                   00000                |
|       11.                           0                   00000                |
|       12.                           0                   00000                |
|       13.                           0                   00000                |
|       14.                           0                   00000                |
|       15.                           0                   00000                |
|       16.                           0                   00000                |
|       17.                           0                   00000                |
|       18.                           0                   00000                |
|       19.                           0                   00000                |
|       20.                           0                   00000                |
|_____________________________________________________________________________|
```

Figure 33. Example archive edit screen.

```
CASE NAME: ATC                    ARCHIVE DATABASE SCREEN              LAST UPDATED: 890727
ARCHIVE SET NAME: TEST1                                               RECORD NUMBER: 000033
|-----------------------------------------------------------------------------------------|
|1. TITLE: ATC TEST AS IN APPENDIX A          2 STUDY:  TEST CASE FOR NEW VERSION         |
|                                                                                         |
|                .             3 MODEL NAME: ATCTEST           4 DURATION:    1.000 HOURS  |
|5 RUN DATE: 10/27/93          8 ENGINEER:   J. KNOX           9 TIME STEP: .16670  HOURS  |
|                                                                                         |
|10 PURPOSE: TO TEST ARCHIVE FOR VERSION 5.0.                                             |
|                                                                                         |
|                                                                                         |
|11 MODEL DESCRIPTION:  ATC IS A SIMPLIFIED MODEL SIMULATING AN ACTIVE THERMAL CONTROL LOOP.|
|                                                                                         |
|                                                                                         |
|12 RESULTS: THE RESULTS OF THE SOLUTION COMPARE FAVORABLY TO THOSE IN APPENDIX A.        |
|                                                                                         |
|                                                                                         |
|13 RUN SPECIFIC   THE MODEL PARAMETERS ARE EXACTLY THE SAME AS THOSE IN APPENDIX A.      |
| MODEL PARAMETERS:                                                                       |
|                                                                                         |
|                                                                                         |
|                                                                                         |
|                                                                                         |
|_________________________________________________________________________________________|
```

Figure 34. Example archive data base screen.

6.8 Archive Process

CASE/A gives the user the option to save or archive any simulation solution by using the archive process. To set up the archive process prior to running a solution for a case and using the ARCHIVE command, refer to section 6.7. During the archive process, the data in the case ".MOD" file, and the control, labels, usercon, plot, and archive data bases are copied to an ASCII file along with the OPS code, if any. The user has the option to archive data associated with the plotsets previously selected in the setup process for archiving. If this option is selected, the ".DAT' and ".STT" files are copied to the archive subdirectory. The user has the option to select the directory where the archived files are stored.

The convention for storing the archive is to have a directory named "ARCH" in the user's CASE/A directory (the path would be "device:[username.CASEA.ARCH]." Alternatively, the user may also specify another existing ARCH directory as an option in the Archive process. If the user does not specify a preexisting subdirectory, the archive process will create a subdirectory of the current directory and name it [.ARCH]. The user can copy older [.ARCH] files to this subdirectory and use the RETRIEVE command as necessary to access these files.

The archive process begins by entering the ARCHIVE command. The dialog in figure 35 shows the messages and responses described here. Next, select the archive directory by entering the path of the directory or hitting enter to select the default directory shown. Active archive sets are then verified in the directory chosen. Next, enter "yes" to export the current case or specify "no" followed by the case to be exported. The case data are exported/copied to the archive directory files as shown. Finally, enter "yes" if desired to archive the plot files generated when the solution was run.

Note that if OPS code does not exist for the case being archived, this message will be displayed:

```
'OPS CODE DOES NOT EXIST FOR THIS CASE.'
```

A CASE/A prompt is displayed when the archive process is terminated.

6.9 Retrieve Archive Model

A model can be retrieved from an archive file using the RETRIEVE command. Section 6.8 describes the conventions and process for creating a model archive. The user begins by creating a new case (NEWCASE command) into which the archived model will be merged. Then, at the CASE/A prompt, enter the RETRIEVE command. Next, select the directory which contains the archive files. A listing of archive files from the directory is presented. Select the source case name (for example MB6_X from MB6_X_EXAMPLE2) and the archive set name (for example EXAMPLE2 in MB6_X_EXAMPLE2) of the archive file. Next, select 1 of 3 options for merging the archive file as follows:

(1) Subsystems, Control, ... (i.e., the entire model)

(2) Subsystems only

(3) Control, Labels, Plot, Usercon, and archive (everything but subsystems)

(If subsystems are selected, the user may select 1 or more of the subsystems or select them all). At this point the archive source/set is merged into the new case. Finally, the user is provided with options to selectively discard parts of the archive model (such as Labels, Control, Usercon, Plot, and Archive records and Ops FORTRAN code) so that these records are not incorporated into the new case. The newcase now contains the parts selected from the archived model and is now ready for modification for the user's purposes or running a solution (provided the user selected all the necessary parts in the above process).

```
CASEA > LOADCASE ATC
CASEA_ATC > ARCHIVE

THIS UTILITY WILL COPY YOUR MODEL, IPU PLOT DATA (IF DESIGNATED FOR ARCHIVING),
AND ASSOCIATED OPS CODE (IF ANY) TO A STORAGE AREA FOR ARCHIVING.
PRESS <RET> TO CONTINUE OR -1 TO ABORT.

ENTER ARCHIVE DIRECTORY NAME (DEFAULT IS [NAIDEAJ.SV.ARCH])
OR PRESS <RET> TO CONTINUE.
DEFAULT DIRECTORY WILL BE CREATED IF IT DOES NOT EXIST.

CHECKING FOR ACTIVE ARCHIVE SETS...

MERGING MODEL TO ARCHIVE FILE...
Export Current Case? [Y/N] > Y

LOCATING TARGET CASE MODEL DEFINITION DATA...

LOADING TARGET CASE EQUIP DATA INTO MEMORY...

LABELS ASCII OUTPUT COMPLETE.

CONTROL ASCII OUTPUT COMPLETE.

USERCON ASCII EXPORT COMPLETE.

PLOT ASCII EXPORT COMPLETE.

COPYING ARCHIVE DATA TO ARCHIVE DIRECTORY...

ARCHIVE ASCII EXPORT COMPLETE.

CHECKING FOR ACTIVE PLOTSETS...


The final query displayed is the following:

DO YOU WANT TO ARCHIVE THE DATA ASSOCIATED WITH THE SELECTED PLOTSETS(Y/N)?
(THIS WILL CREATE TWO NEW FILES FOR EACH PLOTSET): Y

COPYING IPU PLOT DATA TO ARCHIVE FILE...
%COPY-S-COPIED, USER3$:[NAIDEAJ.SV]ATCPLOT.DAT;12 copied to
USER3$:[NAIDEAJ.SV.ARCH]ATC_TEST1_ATCPLOT.DAT;7 (12 blocks)
%COPY-S-COPIED, USER3$:[NAIDEAJ.SV]ATCPLOT.STT;12 copied to
USER3$:[NAIDEAJ.SV.ARCH]ATC_TEST1_ATCPLOT.STT;7 (12 blocks)

COPYING OPS CODE TO ARCHIVE FILE...

ARCHIVE OPERATION FINISHED.

CASEA_ATC >
```

Figure 35. Example archive process.

7.0 USER OPERATIONS LOGIC

The CASE/A system supports the addition of user-defined computations into the solution process. This capability is consistent with other large scale system simulation programs such as SINDA (VARIABLES1, VARIABLES2) and G189A (GPOLY1, GPOLY2). The user may perform custom initialization and wrap-up functions, time varying operations, calculations within the iterative solution at each time step, and create custom output data. Special components such as the TIMER and CNTRLLR support user-defined control operations of all other CASE/A components.

Custom components may be created by use of the BLACKBOX component and associated logic. The addition of custom logic is supported via eight subroutine entry points that are called at various locations in the solution process. They have been named OPS0 through OPS7 for easy identification. These OPS subroutines must be written in FORTRAN (the language in which CASE/A was developed). They are then compiled and linked with the CASE/A routine library to create a customized version of CASE/A. See sample problem 2 in appendix A for a listing of example OPS logic.

7.1 OPS Logic Structure

There are seven subroutines provided for solution control logic and one subroutine for the BLACKBOX component logic. The seven solution control logic subroutines are named OPS0 through OPS6 according to the relative locations of the routine calls in the solution process (fig. 36).

The seven control OPS routines are discussed below:

1. OPS0 is called just before the component initialization calculations are performed. It is useful for presimulation initialization operations such as setting user-defined summation variables to initial values or opening a special file to receive user output. Any conditional modifications of data base parameters needed before the component initializations are performed should be placed here. The user will usually invoke the SEQUENCE routine in this block if alteration of the pseudo compute sequence is desired (see section 7.3).

2. OPS1 is similar to the OPS0 routine, but it is called just after the component initialization calculations are performed. Any custom operations that depend on the initial conditions of the components in the model should be placed here.

3. OPS2 is called just before the iteration process begins for each time step. It is useful for imposing time-varying functions on the system or any operation that needs to be performed before each time step.

4. OPS3 is called every system iteration before the component routines are called. Logic placed here will be executed a variable number of times each time step depending on the number of system iterations required for convergence. A specialized debugging code may be placed here to monitor successive iteration results.

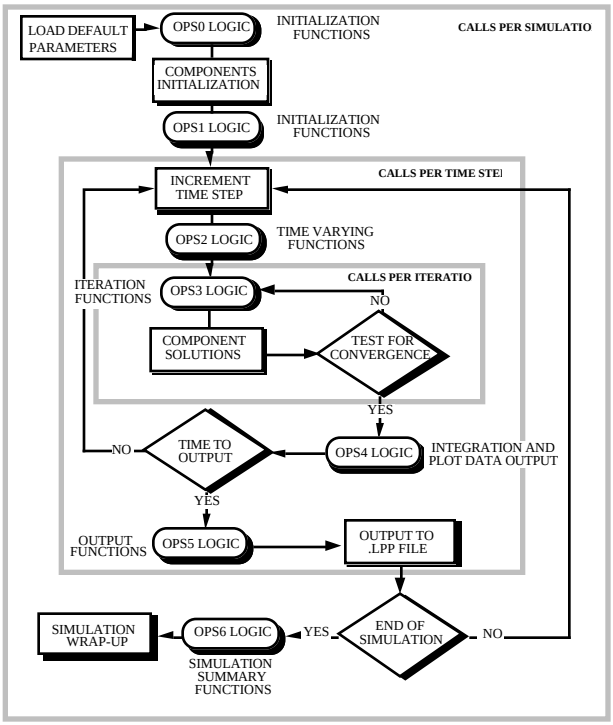


Figure 36. OPS routine calls in solution flow.

5. OPS4 is called just after system convergence has been obtained, but before the time has been incremented for the next time step. It is useful for performing integration functions and writing out custom plot data.

6. OPS5 is called just before the output summary file CASENAME.LPP is updated at each output interval. It is useful for coordinating custom output functions with the system output time interval.

7. OPS6 is called just after the final time step has been completed, but before all the component routines perform their final wrap-up operations. It is useful for post simulation summaries.

The subroutine OPS7 is associated with the BLACKBOX component and is called in conjunction with the other component routines (see section 7.6.3).

7.2 System Data Storage and Manipulation

The CASE/A program stores the component performance parameters, fluid compositions, fluid thermodynamic properties, simulation control variables, custom user constants, and component internal storage parameters in various internal arrays. These arrays may be accessed from the OPS logic blocks via subroutines that interface with the internal data structures. The addressing of components is keyed to the component name given at the time of system generation. Care should be exercised to assure uniqueness of the component names to allow effective use of the data access routines. The data arrays and access routines are described in the applicable sections below.

7.2.1 Component Performance Parameters (CON Array)

The component performance parameters are loaded from the data base files into the internal CON array during the initialization phase of the simulation. The performance parameters are stored contiguously in the CON array, which is currently set to 75,000 total storage locations. Each of the components in a model reserves a storage block in the CON array based on the number of performance parameters set aside for that particular component type. This number varies widely for the equipment types supported in the CASE/A program. The layout of the CON array is illustrated in figure 37.

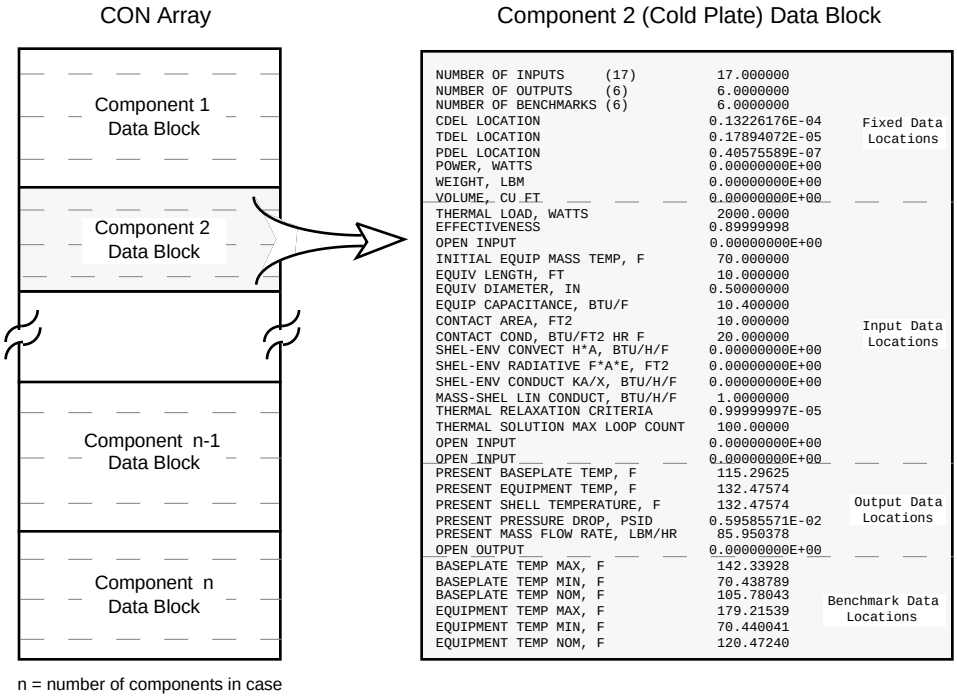


Figure 37. CON array layout.

Each of the component data blocks are divided into four separate sections as described in section 3.1 and repeated here:

1. Fixed data section (always set to 9 spots)

2. Input data section (variable length)

3. Output data section (variable length)

4. Benchmark data section (variable length).

Easy access to all of the parameters is provided by the GETK and SETK routines as described in table 7. The data section in which a specific parameter is contained does not affect its access from within the OPS routines. Only the relative location within the whole component data block is used in the access routines to reference a specific performance parameter. For example, the thermal load of the cold plate (shown in fig. 37) is referenced as location 10 since it is the tenth entry in the cold plate data block. The location numbers of the performance parameters are shown in the component edit screens to the left of each parameter label (see sections 3 and 10).

NOTE: Line editor users must subtract 3 from the item numbers shown on the listing of the data base values to determine the relative CON array locations. This offest is due to the internal storage method used by CASE/A.

7.2.2 Stream Fluid Composition (C Array)

The CASE/A program tracks the fluid mass flow rate and composition of each of the component streams in the C array. This array stores the total mass flow rate of the fluid for flow streams (total mass for accumulator streams) in location 1 and the mass fractions of the constituents in locations 2 to 50. The fluid constituents layout is shown in table 8. Easy access to all of the constituents is provided by the GETC and SETC routines as described in table 9.

Table 7. CON array data access routines.

Subroutine:	**GETK**
Format:	`CALL GETK('NAME....',LOCATION,VALUE)`
Function:	This routine retrieves the value of a component performance parameter. The NAME argument is the alphanumeric name assigned to the component. It must be enclosed in quotes and should be 8 characters, left justified, with trailing blanks. The LOCATION argument is an integer variable which represents the relative location of the parameter in the component data base. The parameter value is returned in the real variable VALUE.
Subroutine:	**SETK**
Format:	`CALL SETK('NAME....',LOCATION,VALUE)`
Function:	This routine modifies a specified component data base parameter. The arguments are as defined for the GETK routine. The VALUE argument represents the value to which the specified data base parameter will be set.

Table 8. Fluid constituents definition.

Location	Content	
1	Total Mass Flow Rate, Lb/Hr	(Mass for Accumulators, Lb)
2	Oxygen	Mass Fraction
3	Nitrogen	Mass Fraction
4	Carbon Dioxide	Mass Fraction
5	Hydrogen	Mass Fraction
6	Solid Carbon	Mass Fraction
7	Methane	Mass Fraction
8	Water	Mass Fraction
9	Freon - 11	Mass Fraction
10	Constituent 10	Mass Fraction
11	Constituent 11	Mass Fraction
..	. .	. .
50	Constituent 50	Mass Fraction

Table 9. C array data access routines.

Subroutine: **GETC**

Format: `CALL GETC('NAME....',NSTREAM,NCONSTITUENT,VALUE)`

Function: This routine retrieves a value from the constituent data array. The NAME argument is the alphanumeric name assigned to the component. It must be enclosed in quotes and should be 8 characters, left justified, with trailing blanks. The NSTREAM argument is an integer variable which identifies the component stream number of interest. The stream numbers for each component are shown in section 10. The NCONSTITUENT argument is an integer variable which identifies the specific constituent required. Table 7.2.2-1 identifies the chemical species associated with each constituent number. The constituent mass fraction or flow rate data is returned via the real variable VALUE.

Subroutine: **SETC**

Format: `CALL SETC('NAME....',NSTREAM,NCONSTITUENT,VALUE)`

Function: This routine modifies a constituent value in the constituent data array. The arguments are as defined for the GETC routine. The VALUE argument represents the value to which the specified constituent location will be set.

7.2.3 Stream Fluid Properties (PRO Array)

The solution system tracks the fluid properties of each of the component streams in the PRO array. This array contains six thermodynamic properties of each fluid stream calculated as the mass weighted average values of the constituents present. A listing of the properties is shown in table 10. Easy access to the temperatures and pressures is provided by the GETT, SETT, GETP, and SETP routines as described in table 11.

Access to all properties of a fluid stream may be obtained by addressing the PRO array directly. The PRO array is a three-dimensional array in which the first index indicates the property number, the second index indicates the relative number of the desired component in the pseudo compute sequence, and the third index indicates

Table 10. Fluid properties definition.

Location	Content
1	Pressure, psia
2	Temperature, ˚F
3	Specific Heat (C_p), Btu/lbm/˚F
4	Density, lbm/ft^3
5	Viscosity, lbm/hr/ft
6	Specific Heat (C_v), Btu/lbm/˚F

Table 11. PRO array data access routines.

Subroutine: **GETT,GETP**

Format:
```
CALL GETT('NAME....',NSTREAM,VALUE)
CALL GETP('NAME....',NSTREAM,VALUE)
```

Function: These routines retrieve temperatures (GETT) and pressures (GETP) of the component fluid streams. The NAME argument is the alphanumeric name assigned to the component. It must be enclosed in quotes and should be 8 characters, left justified, with trailing blanks. The NSTREAM argument is an integer variable which identifies the component stream number of interest. The stream numbers for each component are shown in section 10. The data is returned in the real variable VALUE.

Subroutine: **SETT,SETP**

Format:
```
CALL SETT('NAME....',NSTREAM,VALUE)
CALL SETP('NAME....',NSTREAM,VALUE)
```

Function: These routines modify the temperature and pressure data for a specified component. The arguments are as defined for the GETT and GETP routines. The VALUE argument represents the value to which the temperature or pressure will be set.

the identifying number of the stream. For instance, the densities of streams 1 and 2 of a FILTER component are referenced as follows:

$$\rho_{in} = \text{PRO}(4,\text{IEQ},1)$$
$$\rho_{out} = \text{PRO}(4,\text{IEQ},2)$$

where,

IEQ = Relative component number in pseudo compute sequence (returned by routine GETI described in table 14).

7.2.4 Internal Simulation Control Variables

The user may access the internal simulation control variables such as the current simulation time, time step, or the number of constituents tracked in a model from within the OPS routines. The simulation control variables are contained in individual variables that are grouped together into a single common block in memory named EXQ. A listing of the variables contained in this memory block is shown in table 12. The

Table 12. Internal simulation control variables.

Name	Description
TIME	The current simulation time in hours.
STRT	The simulation starting time in hours, minutes, or seconds depending on the time unit flag (defaults to 0.0).
END	The simulation termination time in hours, minutes, or seconds depending on the time unit flag (defaults to 24.0).
STEP	The simulation time step interval in hours, minutes, or seconds depending on the time unit flag (defaults to 1.0).
RELX	The relative convergence criterion to be met (defaults to 1.0E-4).
IEQ	The current relative equipment number in the pseudo compute sequence.
LPCS	The current relative connection pointer in the pseudo compute sequence.
MFLAG	Indicates the code segment to execute in the component routines.
RXCC	Tracks the maximum relative change of all component streams in the case.
IPTFLG	Indicates whether the user wishes the system to perform relaxation checks on temperatures and/or pressures. The default value of 0 indicates checks are to be made for both. A value of 1 indicates that the pressures are not to be checked while a value of 2 indicates that the temperatures are not to be checked.
NEQ	The total number of components in the case to be simulated.
NCON	The total number of connections in the case to be simulated.
NCOMP	The number of component types currently supported by CASE/A.
NTRACK	The number of constituents to be tracked (up to 50).
NSYSLOOP	The current number of system iterations performed during the present time step.
SYSDAMP	A damping factor used by a few specialized hydraulic routines.
NSYSMAX	The maximum number of iterations to perform during the iteration process.
SYSOUTPT	The simulation output interval in hours, minutes, or seconds depending on whether ITIMFLG is set to 1, 2 or 3.
NSYSITER	The current total number of system iterations performed.
ITIMFLG	The time unit flag indicates whether the user wished the simulation time messages to be reported in hours (1), minutes (2), or seconds (3). This is entered in the ED CONTROL screen.
UCONFAC	The unit conversion factor for the simulation times (start, step, and end) based on the ITIMFLG (time unit flag).
IUNITS	The simulation output interval units which can be hours, minutes, or seconds.

user must ensure that this common block is declared at the beginning of each OPS routine in which access to the control variables is required. This may be accomplished by the inclusion of the following statement as the second line of each routine:

```
INCLUDE 'CASEA$CODE:COMPCOM.INC/LIST'
```

This statement refers to a file in the CASE/A program code directory that contains the common block memory declarations. It must be in each routine that needs access to the simulation control variables. The skeletal OPS routines that are supplied in the file OPS.FOR with the CASE/A program contain this statement. See sample problem 2 in appendix A and the installation notes in appendix C for further explanation.

The user must take care in using these variables in the OPS routines. They must not be reset to erroneous values since the simulation is controlled by them. Unpredictable results may occur if extreme caution is not exercised.

7.2.5 Custom User Constants

The CASE/A program supports the interactive specification of 60 custom user constants and provides data access routines to utilize them in the OPS routines. They are loaded from a data base file into the single-dimensional USERCON array (real data only) at the beginning of a simulation. The user may edit them interactively as follows:

```
ED;USERCON
```

This command invokes the standard CASE/A editor in either the screen or line edit modes as determined by the terminal device type (see section 3). The user may specify an alphanumeric label up to 24 characters in length for each of the constants as shown in figure 38.

The user may access the constants from within the OPS routines with the GETU and SETU routines as described in table 13. The final values of the user constants are written back to the data base file for permanent storage at the completion of a simulation. Therefore, summation or benchmark values of interest to the user may be created and examined. The user may view the results by editing the USERCON data base after the completion of a simulation.

7.2.6 Component Internal Parameters (D Array)

Several of the component routines store data for internal usage or for special debugging purposes in the two-dimensional D array. The first index indicates the relative number of the component in the pseudo compute sequence (PCS), and the second index indicates the location number in the D array. The CASE/A program sets aside 50 locations in the D array for each of the components in a model. The user may access these locations only by addressing the D array directly as follows:

$$D_1 = D(IEQ,1)$$
$$D_2 = D(IEQ,2)$$

```
CASE NAME: DEFAULT                    USER DEFINED CONSTANTS EDIT SCREEN                    LAST UPDATED: 891011
                                                                                          RECORD NUMBER: 000001
| 1 ..................  0.0000E+00 | 21 ..................  0.0000E+00 | 41 ..................  0.0000E+00 |
| 2 ..................  0.0000E+00 | 22 ..................  0.0000E+00 | 42 ..................  0.0000E+00 |
| 3 ..................  0.0000E+00 | 23 ..................  0.0000E+00 | 43 ..................  0.0000E+00 |
| 4 ..................  0.0000E+00 | 24 ..................  0.0000E+00 | 44 ..................  0.0000E+00 |
| 5 ..................  0.0000E+00 | 25 ..................  0.0000E+00 | 45 ..................  0.0000E+00 |
| 6 ..................  0.0000E+00 | 26 ..................  0.0000E+00 | 46 ..................  0.0000E+00 |
| 7 ..................  0.0000E+00 | 27 ..................  0.0000E+00 | 47 ..................  0.0000E+00 |
| 8 ..................  0.0000E+00 | 28 ..................  0.0000E+00 | 48 ..................  0.0000E+00 |
| 9 ..................  0.0000E+00 | 29 ..................  0.0000E+00 | 49 ..................  0.0000E+00 |
| 10 .................  0.0000E+00 | 30 ..................  0.0000E+00 | 50 ..................  0.0000E+00 |
| 11 .................  0.0000E+00 | 31 ..................  0.0000E+00 | 51 ..................  0.0000E+00 |
| 12 .................  0.0000E+00 | 32 ..................  0.0000E+00 | 52 ..................  0.0000E+00 |
| 13 .................  0.0000E+00 | 33 ..................  0.0000E+00 | 53 ..................  0.0000E+00 |
| 14 .................  0.0000E+00 | 34 ..................  0.0000E+00 | 54 ..................  0.0000E+00 |
| 15 .................  0.0000E+00 | 35 ..................  0.0000E+00 | 55 ..................  0.0000E+00 |
| 16 .................  0.0000E+00 | 36 ..................  0.0000E+00 | 56 ..................  0.0000E+00 |
| 17 .................  0.0000E+00 | 37 ..................  0.0000E+00 | 57 ..................  0.0000E+00 |
| 18 .................  0.0000E+00 | 38 ..................  0.0000E+00 | 58 ..................  0.0000E+00 |
| 19 .................  0.0000E+00 | 39 ..................  0.0000E+00 | 59 ..................  0.0000E+00 |
| 20 .................  0.0000E+00 | 40 ..................  0.0000E+00 | 60 ..................  0.0000E+00 |
```

Figure 38. User constants default edit screen.

Table 13. User constants data access routines.

Subroutine:	**GETU**
Format:	CALL GETU(LOCATION,VALUE)
Function:	This routine retrieves a value from the user constants array. The LOCATION argument is an integer variable which identifies the relative number of the constant (60 constants are available). The value of the constant is returned in the real variable VALUE.
Subroutine:	**SETU**
Format:	CALL SETU(LOCATION,VALUE)
Function:	This routine modifies a user constant value. The arguments are as defined for the GETU routine. The VALUE argument represents the value to which the specified user constant will be set.

where,

IEQ = Relative component number in pseudo compute sequence (returned by routine GETI described in table 14).

The user must be familiar with the component routine code to utilize the D array data. See the CASE/A programmer's manual for more detail on its utilization.

7.3 Component Solution Order Sequencing

The pseudo compute sequence (PCS) defines the order in which the component subroutines are called in the solution process. The PCS is initially set to the order in which the components are located in the system schematics in the model construction phase. The user may modify the PCS by invocation of the SEQUENCE routine from the OPS logic routines (shown in table 14) to optimize the mass transfer calculations to proceed in a flow direction sequence.

Table 14. PCS interface routines.

Subroutine:	**SEQUENCE**

Format:	`CALL SEQUENCE('SEQSTART')` ! Initiates the list
	`CALL SEQUENCE('NAME_1..')`
	`CALL SEQUENCE('NAME_2..')`
	`CALL SEQUENCE('SEQSTOP ')` ! Terminates the list

Function:	This routine allows the user to specify a group of components in a sequential list as they are to be invoked by the solution system. The remainder of the components in the case are invoked in the order in which they were input into the graphical schematics. The NAME argument is enclosed in quotes as indicated and should be eight characters, left justified, with trailing blanks. The SEQUENCE calls are placed in the OPS0 routine if an initial sequence is desired or in the OPS2 routine for time-dependent resequencing.
Subroutine:	**GETI**
Format:	`CALL GETI('NAME....',IVALUE)`
Function:	This routine retrieves the pseudo compute sequence pointer for the indicated component. The NAME argument is the alphanumeric name assigned to the component. It must be enclosed in quotes and should be eight characters, left justified, with trailing blanks. The pointer is returned via the integer IVALUE argument. The pointer may be used for special logic or in reference to the component PRO or D arrays.

7.4 OPS Logic Incorporation

The user OPS routines must be linked with the rest of the CASE/A program to create a customized version incorporating the user operations. A logical way to accomplish this is to include all of the OPS routines in a CASENAME.FOR file. This file must be compiled with the FORTRAN compiler and linked with the CASE/A library modules to create an executable image called CASENAME.EXE. The custom CASE/A environment acts identically to the generic version except that the solution flow will be altered by the user control operations.

The steps shown in figure 39 are performed when the following commands are entered to the VAX operating system:

1.	COPY OPS.FOR	CASENAME.FOR
2.	EDIT	CASENAME.FOR
3.	FORTRAN	CASENAME
4.	@LINKCAS	CASENAME
5.	RUN	CASENAME

Template OPS routines are provided with the CASE/A system in a file named OPS.FOR. The user may copy this file into a model-unique file named CASENAME.FOR. The model-specific logical or computational operations may then be incorporated into the appropriate FORTRAN subroutines with the VAX editor.

The FORTRAN command converts the CASENAME.FOR file into machine language instructions contained in the CASENAME.OBJ file. This OBJ file must then be linked with the CASE/A program library. A macro command file named LINKCAS.COM has been provided for this purpose. It should be located on the user working directory and is invoked as indicated. It will perform the linkage process to create an executable image file named CASENAME.EXE. This file may then be executed to enter the CASE/A environment. The user logic will take effect when the solution command is issued.

The user must repeat the incorporation process if component deletions are performed on the model or if the OPS logic is updated. The custom executable file will correctly solve only the model to which it applies.

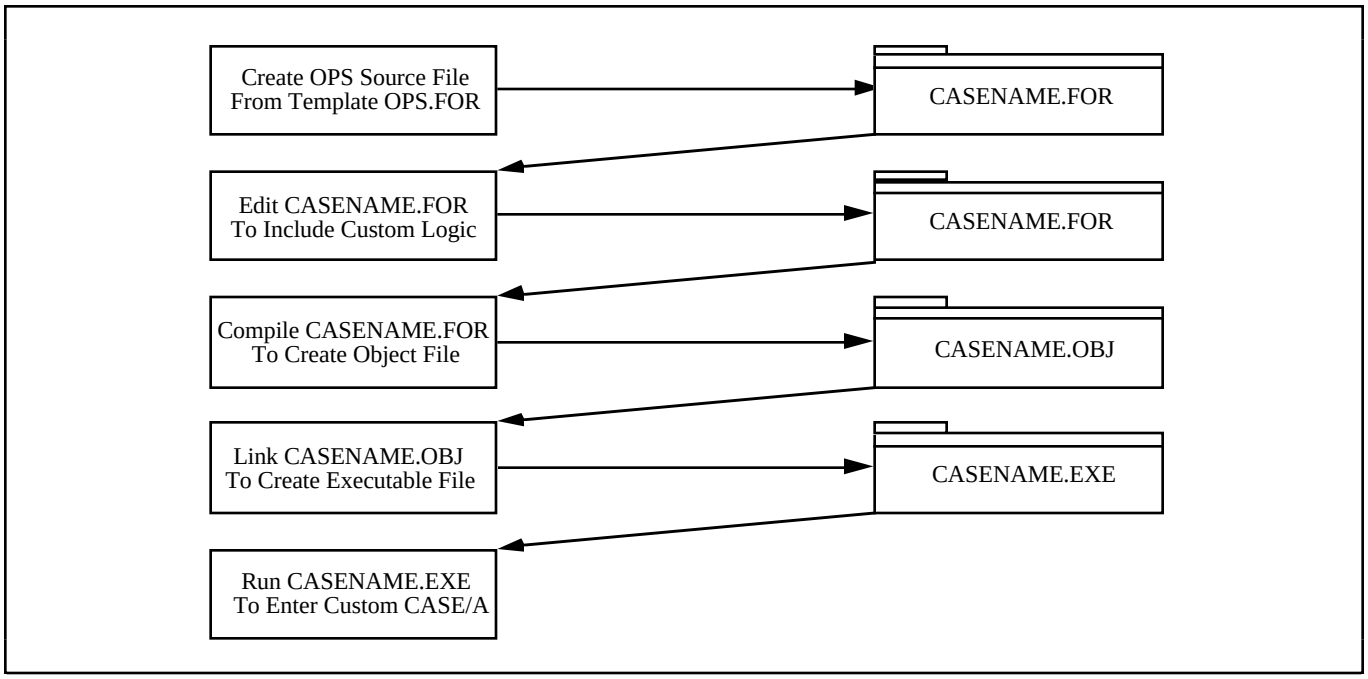


Figure 39. OPS logic incorporation procedure.

7.5 OPS Logic Preprocessor

An alternate mechanism has been provided to retrieve and modify system variables in the PREPRO function. In large models and/or long simulations, the use of the data access routines can become time consuming, since they search through the entire case to locate a referenced component. The PREPRO function resolves the data references before the solution is initiated so that the data arrays are addressed directly. It requires the user to exit the program completely since it generates a new OPS source file that explicitly addresses the data array locations in the source code. The user must then recompile and relink the resulting OPS file with the CASE/A program as described in section 7.4.

The PREPRO command requires the creation of a file named CASENAME.OPS before it is invoked as follows:

```
PREPRO;CASENAME.OPS
```

This file contains special formatting of the data access requests (routines such as GETC, SETC, and GETK are not used) but which is otherwise identical to a standard OPS file. The preprocessor will convert a properly formatted CASENAME.OPS file into a preprocessed file named CASENAME.FOR. This CASENAME .FOR file may then be compiled and linked with the CASE/A program library to create an executable image file.

The format guidelines for the CASE NAME.OPS file are shown in table 15. The user will be informed, by a message to the screen, of the success of the preprocessing operation. The preprocessor will detect format errors, missing delimiters, and improper equipment names and will write an appropriate error message in the CASENAME.FOR file. Errors can be corrected in the CASENAME.OPS without leaving the program by use of the EDT command, which activates the VAX editor. After successful preprocessing, the user must exit the CASE/A program and perform the appropriate VAX compilation and linkage commands to create the CASENAME.EXE file as described in section 7.4. It is then run in order to solve the CASENAME model with the newly created preprocessed OPS logic file.

The user must repeat the entire PREPRO process if component additions or deletions are performed or if the OPS logic is updated. The user must understand that the custom EXE file will correctly solve only the specific model to which it applies.

Table 15. Format guidelines for the PREPRO routine.

Desired Quantity	Format	Resulting Code
Stream Pressure	[P;EQNAME:STREAM NO.]	PRO(1,IEQ,STREAM NO.)
Stream Temperature	[T;EQNAME;STREAM NO.]	PRO(2,IEQ,STREAM NO.)
Stream Heat Capacity	[H;EQNAME;STREAM NO.]	PRO(3,IEQ,STREAM NO.)
Stream Density	[D;EQNAME;STREAM NO.]	PRO(4,IEQ,STREAM NO.)
Stream Viscosity	[V;EQNAME;STREAM NO.]	PRO(5,IEQ,STREAM NO.)
Component Data base Parameter	[K;EQNAME;DATABASE LOCATION]	CON(#)
Total Mass Flow Rate	[M;EQNAME;STREAM NO.]	C(1,#)
Constituent Mass Fraction	[C;EQNAME;STREAM NO.; CONSTIT NO.]	C(CONSTITUENT NO.,#)
Relative Component No.	[I;EQNAME]	IEQ

Where; IEQ = Relative Component Number
= Appropriate Array Value Calculated by Preprocessor

7.6 Special CASE/A Components

The CASE/A program supports two control components that are very useful for simple control tasks: TIMER and CNTRLLR. Their control operations are limited in scope and number, but they may be interactively modified from within the CASE/A runtime environment. This capability allows the user to incorporate simple control functions without resorting to OPS logic generation. The BLACKBOX component is provided to allow the advanced user to easily create custom components. These three components are discussed briefly in the following sections and in the component descriptions in section 10.

7.6.1 TIMER Component

The TIMER supports the control of multiple parameters in the performance data block of any CASE/A component based on timelined input data. The timeline data may be a list of discrete timed events relative to the simulation time or a list of events to be repeated on a set cycle period.

The TIMER may be placed on the schematic subsystem screen containing the component to be controlled. A connection data path is then established between the TIMER hit box and the body hit dot of the component to be controlled with the `CN` command. The BUBBLE interconnect feature may be utilized to bridge across subsystem screens. The interconnecting data path is drawn as a dashed line to distinguish it from a normal fluid connection line. The connection process is shown in figure 40.

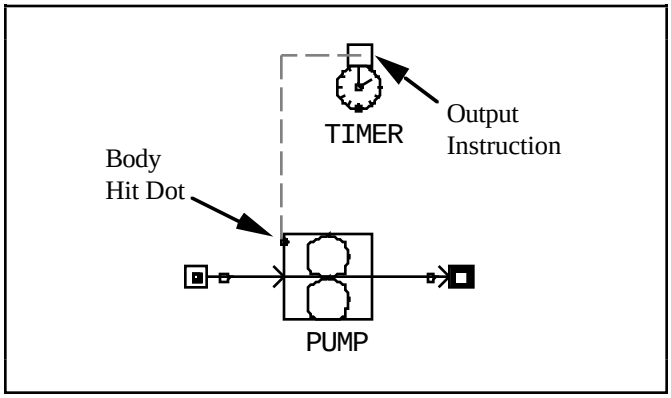


Figure 40. Example TIMER connection.

7.6.2 CNTRLLR Component

The CNTRLLR provides the capability to monitor and control component data such as the fluid stream properties, constituents, and performance parameters as well as certain system control variables. Basic controller actions such as two-position control and proportional control are included as built-in options. The CNTRLLR reverse polish notation programming language may be utilized to custom design user control logic.

The CNTRLLR "streams" represent data flow paths that gather and transmit data and control instructions. They may be connected to either the body or stream data hit dots of the components involved in the control process as shown in figure 41. The CNTRLLR is located and connected in a similar manner as the TIMER component. In the example shown, the CNTRLLR will gather an input signal from the PUMP component, such as the inlet fluid temperature. It will then perform the internal control calculations based on the inlet data value, such as proportional adjustment of the PUMP mass flow rate to achieve a set inlet temperature. The output instruction will then be sent to the PUMP, such as setting the mass flow rate in the performance parameters block equal to the internally calculated value. The PRT instruction may be utilized as a debugging tool to write various parameters to the display device.

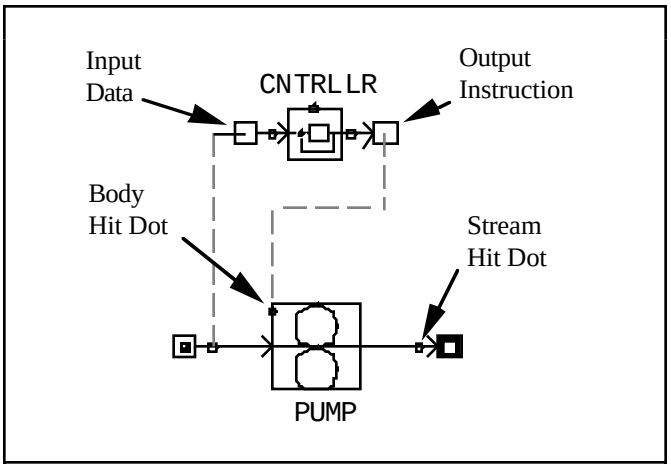


Figure 41. Example CNTRLLR connection.

7.6.3 BLACKBOX Component

The BLACKBOX allows the user to create specialized components/processes not currently modeled by the CASE/A system. By using this feature, system level support of items such as graphical icons, stream connections, solution calls, etc., is provided automatically. A maximum of eight fluid streams is allowed for the BLACKBOX component with up to four inlets and four outlets. A total of 25 BLACKBOX components may be used in a model. These may be the same type of component/process, all different types, or any combination. The user is responsible for providing the FORTRAN logic required to simulate the process under consideration. A template routine, named BBOX.FOR, is provided with the CASE/A package and is located in the code directory. Each different type of BLACKBOX requires its own code. As an example, suppose that a model has five BLACKBOX components, three of one type, and one each of two other types (a total of three types). Thus, three copies of BBOX.FOR must be copied and coded (each must also be uniquely named). For convenience in compiling and linking, these routines may be appended to the CASENAME.FOR (OPS logic) file. The next step is to modify subroutine OPS7 (located in the CASENAME.FOR file). Continuing with the above example, a modified OPS7 file is shown in figure 42. It is assumed that the three specialized routines were named BBOX1, BBOX2, and BBOX3. Whenever a BLACKBOX (any type) is encountered during the solution process, OPS7 is called. This routine will then call the appropriate specialized routine, depending on which type of BLACKBOX is being solved. Note that if numerous BLACKBOX components are developed, the same OPS7 file can be used in different CASENAME.FOR files. Suppose a second model is developed which requires two BLACKBOX components of a type already coded, say BBOX2. The same OPS7 file could be used, and when the BLACKBOX components are located on the subsystem screen, they are designated as type 2. As can be seen, the types of BLACKBOX components in a given model do not need to start at “1” nor do the types need to be numbered consecutively. See sections 7.1, 8.1, and 10.4 for further details.

```
      SUBROUTINE OPS7
      INCLUDE 'CASEA$CODE:GRAPHCOM.INC'
C
      DO 10 I = 1, 25
   10   IF(NBOX(I,3) .EQ. IEQ) NBBTYPE = NBOX(I,4)
C
      GOTO (100,200,300,400,500,600,700,800,900,1000,
     &     1100,1200,1300,1400,1500,1600,1700,1800,1900,2000,
     &     2100,2200,2300,2400,2500) NBBTYPE
C
  100 CALL BBOX1 ! CALL BLACK BOX TYPE 1
      GOTO 999
C
  200 CALL BBOX2 ! CALL BLACK BOX TYPE 2
      GOTO 999
C
  300 CALL BBOX3 ! CALL BLACK BOX TYPE 3
      GOTO 999
C
  400 CONTINUE ! CALL BLACK BOX TYPE 4
      GOTO 999
   .
   .
   .
 2500 CONTINUE ! CALL BLACK BOX TYPE 25
      GOTO 999
C
  999 CONTINUE
      RETURN
      END
```

Figure 42. Modified subroutine OPS7.

8.0 ANALYTICAL TECHNIQUES

This section describes typical solution techniques that are utilized throughout the CASE/A system. An overview of the pressure solution methodology, descriptions of the thermal and mass transfer networks, and the fluid properties calculation methods are described. The CASE/A system and documentation use architectural concepts and terminology that originate in the SINDA and G189 analytical computer programs.

8.1 Iterative Solution Process

The CASE/A system performs a quasi-steady state iterative solution at each time interval according to the control parameters specified in the CONTROL data file for the selected case. These control parameters are discussed in detail in section 4.1. The solution iterates on a component level until convergence of the fluid mass flows and compositions, temperatures, and pressures of the system is obtained.

The iterative process begins after the component data initialization at the simulation start time and is continued until the termination time is reached. At each time interval, the main solution routine invokes each of the component routines sequentially according to the internal PCS until all of the components in the case have been processed. After each complete system iteration, the convergence flag indicators for all component streams are checked for compliance with the desired convergence criterion:

$$\lambda = \text{abs}\left(\frac{x^{n+1} - x^{n}}{x^{n+1} + x^{n}}\right),$$

where,

λ = relative convergence criterion
x^{n+1} = latest iteration value of parameter x
x^{n} = previous value of parameter x .

The iteration process at each time interval is repeated until convergence is obtained for each component stream or until the maximum number of iterations specified by the user is reached. The solution system then performs several post time step wrap-up operations, prints a progress message to the display device, and increments the time according to the time step set by the user in the CONTROL data file. This process is then repeated until the termination time is reached.

Each of the component routines supported by the CASE/A system performs four discrete functions corresponding to the segments of the solution process (fig. 43):

1. Component data initialization at start time
2. Iterative mass and energy balance solution each time step
3. Posttime step wrap-up
4. Postsimulation wrap-up.

Each component routine performs initialization operations at the beginning of the simulation to load performance data into active memory from the data base files. Also, any other internal operations required to initialize the component solution are carried out, such as setting up the benchmark data tracked internal to the component. The benchmark data are described in more detail in section 3.1. This segment is performed only once during the simulation.

The iterative segment of the routines contains the mass flow, thermal, and hydraulic calculations that simulate the physical behavior of the components. Each component inlet stream “accepts” the flow conditions from the upstream component outlet stream. The component routine then performs all appropriate calculations treating these inlet flow conditions as fixed boundary values. The routine sets the outlet stream flow conditions according to the performance parameters and operating characteristics of that particular type of equipment. Thus, the mass flows of the system are passed down the connecting trail of components as defined by the user. A convergence check on the mass flows, temperatures, and pressures of all streams is then performed and execution is returned back to the main solution control routine. This segment is the main body of the component logic and is executed several times during the solution of each time interval until system convergence is obtained.

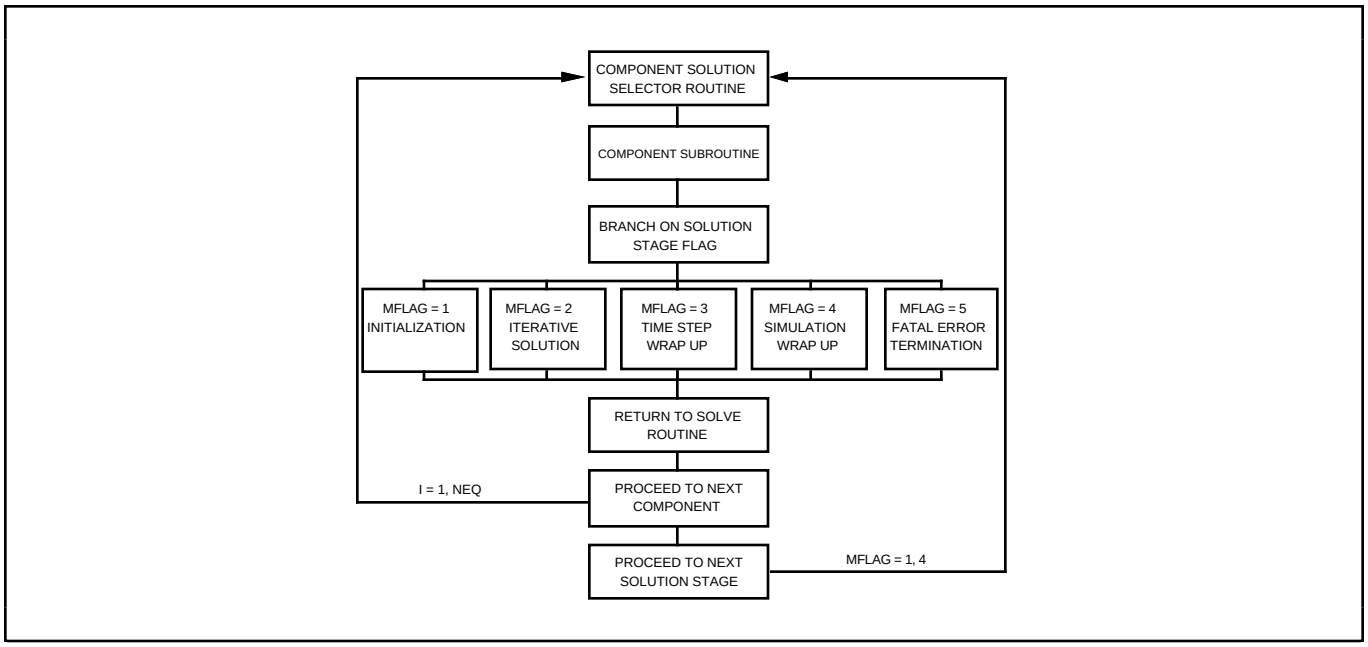


Figure 43. Component logic functions.

After the system has obtained convergence for the present time step, each component performs a post time step wrap-up in which mass accumulation devices perform specialized calculations, and the benchmark data are updated for all components. This segment is executed once for each time interval of the simulation.

After the termination time has been reached, each component performs a post simulation wrap-up where the final benchmark data are updated and written back to the mass storage data base file for that component type. This segment is executed only once at the end of the simulation.

8.2 Pressure Drop Solution

This section provides a description of the methodology used in obtaining a system pressure balance. The solution algorithm involves an implicit iteration on the flow rates versus pressure drops. An implicit algorithm is required because of the nature of the pressure and flow balancing process. The flow rates depend on the pressure differentials, which in turn are dependent on the flow rates.

The pressure solution system is responsible for the pressure drop and mass flow balance versus flow resistance calculations on both the component and system levels. There are two different solution schemes available to the user: matrix reduction and iterative feedback. The method to be used in a given case is set in the CONTROL data base described in section 4.1. The user should normally choose the matrix reduction method since it is the more accurate method and requires fewer system iterations to attain pressure convergence.

PLEASE NOTE: If the user finds that the model is not converging for cases with a large number of components, please try the “Feedback” solution method (specified in the CONTROL file). In the past, problems with convergence have been noted where the “Matrix” solution method was used for large cases.

8.2.1 Matrix Reduction Pressure Solution

The default method of determining the system pressures is the solution of a set of simultaneous equations obtained by conservation of mass at each pressure node. The set of equations is set up for each system iteration discussed in section 8.1. The mass flows and compositions are set by the component routines while the pressures are calculated by the system.

For any node *i* the conservation equation is (for no mass accumulation):

$$\sum \dot{m}_{out} - \sum \dot{m}_{in} = 0 \ ,$$

where,

$\sum \dot{m}_{in}$ = flow added at node i

$$\sum \dot{m}_{out} = \sum_{j=1}^{n} GF_{ij} \, [P_j - P_i]$$

GF_{ij} = linearized flow conductor between nodes i and j

P_i = pressure at node i

P_j = pressure at node j

n = number of pressure nodes.

This form of the conservation equation assumes that there is no mass accumulation for any of the variable pressure nodes. The accumulation nodes, such as the STORE and CABIN components, are assumed to be set boundary pressures from the system solution perspective. The individual accumulator routines are responsible for setting the actual pressure values based on the amount of mass in them. Thus, the CABIN and STORE components become boundary endpoints for the flow subnetworks that are connected to them.

The overall hydraulic balance usually requires several iterations in which the components make successive "guesses" at the flow rates based on the system calculated pressures. The linearized flow conductor between two nodes is thus a function of the flow rate and pressure difference occurring between them (see section 8.2.4) from the last iteration:

$$GF_{ij} = \frac{\dot{m}_{ij}}{\Delta P_{ij}} \ ,$$

where,

ΔP_{ij} = function ($\dot{m}_{ij}{}^2$, L_{eq} , D_{eq}, etc.) .

The system solution methodology is based on the concept of the subdivision of the entire hydraulic layout of the case into separate subnetworks that are terminated by boundary pressure nodes or sources and sinks. These subnetworks may be connected by shared boundary pressure nodes (i.e., STORE's and CABIN's). A set of simultaneous equations is set up and solved for each of these subnetworks individually. This is done in order to decrease the computational resources that would be required to solve a large set of equations for all nodes in the case. The matrix solution used is the symmetric Cholesky method in which the matrix formulation was derived from the SINDA/SINFLO algorithm described in reference 1. Each of the subnetworks should have at least one boundary pressure, or a singular matrix condition occurs in which no solution is possible. In this case, an arbitrary node in the network will be taken as a boundary node and the rest of the pressures calculated accordingly. The user is notified with a run-time warning if this situation occurs.

8.2.2 Iterative Feedback Pressure Solution

The feedback approach allows the characteristics of a closed flow path to be sensed by the component that initiates the mass flow by direct communication with only the immediately adjacent components. Thus, no global knowledge of the complete flow layout is needed, and the mass pressure balance calculations are made by the components that initiate the mass flows or that split the flow based on the downstream flow resistances. A few system iterations are usually required to achieve the correct pressure balance. The flow configuration and PCS determine how fast the system converges. An optimized PCS causes a very significant increase in execution speed, since fewer iterations are required if the components are called in a "flow direction" sequence. The user may alter the PCS through the use of the SEQUENCE routine (see section 7.3) in the user operations blocks. This routine allows the user to specify a sequential list of components to be called in the specified relative order. The rest of the PCS is not affected. Of course, the user may rearrange the entire PCS if necessary by listing all of the components in the case. The user should place the SEQUENCE calls in the OPS1 block if an initial sequence is desired or in the OPS2 block if time-dependent resequencing is desired.

8.2.3 Flow Conductor Calculations

The pressure solution system consists of component level routines that calculate internal component pressure losses, as well as system level routines that perform the interface operations between the component streams through the flow path connections. The interface routines base their

operations on the stream classification codes discussed in the following paragraphs.

The component streams are classified as one of six general types of flow “devices” according to their mass flow characteristics. There are four general categories of stream types: inlet, outlet, conditional, and stagnant. The inlet and outlet streams may be either active or passive. The default stream types are indicated by the fill patterns of the fluid stream hit boxes of the component icons. The stream types and their graphical representations are shown in table 16.

Table 16. Fluid stream classifications.

Stream Type	Representation
1. Passive Inlet	
2. Active Inlet	
3. Passive Outlet	
4. Specified Outlet	
5. Boundary	
6. Stagnant	None

A passive inlet indicates that the stream accepts the mass flow that is passed to it by the upstream component. An example of this type is the inlet stream of a FILTER. The FILTER inlet pressure is determined by adding the pressure drop of the internal flow path to the corresponding outlet stream pressure. Thus, a flow conductor is established between the inlet stream and its corresponding outlet stream.

An active inlet indicates that the stream generates a flow rate by pulling mass from the upstream component. An example of this type is the inlet stream of a PUMP. The active inlet pressure is set equal to the upstream pressure minus the pressure drop through the flow connection.

A passive outlet indicates that the component passes the flow from a passive inlet stream to the outlet side with a flow conductor between the inlet side and the outlet side. Therefore, its pressure is set equal to the downstream component’s inlet pressure plus the pressure drop through the flow connection. An example of this type is the outlet stream of a FILTER.

The specified outlet indicates that the component contains a flow generation device internal to the component that drives the outlet flow without a direct relationship to any of the inlet streams. Its pressure may be set to a constant value to simulate a regulated process, or a feedback loop may be established by passing the PSPEC routine an argument value of zero. In the latter instance, the pressure will be adjusted to match the downstream flow configuration. An example of the regulated type is the oxygen outlet stream of the static feed water electrolysis (SFWE), that relies on a high-pressure nitrogen source to regulate the reaction chamber pressure. An example of the feedback type is a component that contains an integral positive displacement pump. The user has the option for most of the specified outlet streams to either have the pressure regulated to a set value or to establish a feedback loop. This is accomplished by setting the pressure to a nonzero value or to 0, respectively.

A boundary stream indicates that the stream may either accept flow or pass flow depending on (1) components connected to it or (2) on the pressure balance of the system. The component routine is responsible for any interface operations, feedback loop functions, and the calculation of the stream pressure. Examples of this type are STORE, CABIN, and NODE.

A stagnant stream indicates that no flow can occur to or from the stream under any flow conditions. This type usually indicates a deactivated component or perhaps a closed valve. No components have default streams of this type: they are set at run time by the component routines.

The stream classifications are contained in a system-level data array and may be changed dynamically during the solution process as dictated by the component routines. The stream codes are initialized to the component default values at the beginning of the simulation. They are checked for compatibility in the CONDP interface routine, which is responsible for calculating the pressure drops in the flow connections and performing feedback operations. The routine will also verify that the connections make sense from a hydraulic standpoint. For instance, the connection of two inlet streams to each other is an obvious error, and a warning message is printed to

the display. In this case, the simulation is continued, but the pressure drop for the flow connection is not calculated, and the pressures are left at their previous values. The user may encounter a situation where dynamic conditions create a few solution iterations in which the stream codes are incompatible, but which eventually are resolved. However, a continuation of the warning messages indicates an erroneous connection that cannot be resolved.

8.2.4 Friction Losses Through Conduits

The pressure drop through a closed conduit is calculated by the routine FRICTDP, which is called by several components for internal flow paths and for each flow connection. The FRICTDP routine calculates the pressure drop based on the assumption of an incompressible fluid flowing steadily through a smooth circular conduit in the laminar, transitional, and turbulent regimes. The Darcy formula is used in combination with the hydraulic equivalent length and diameter, the average fluid density, and the average fluid viscosity to calculate the pressure drop. The average fluid properties are used to allow for properties variation across flow connections and through internal component fluid loops.

The pressure drop through a conduit is calculated using the following equation for frictional head loss:

$$\Delta P = \frac{2f\rho L_{eq}\upsilon^2}{D_{eq}g144},$$

where,

ΔP = frictional pressure drop, lbf/in^2

ρ = average fluid density, lbm/ft^3

L_{eq} = hydraulic equivalent length, ft

D_{eq} = hydraulic equivalent diameter, ft

υ = fluid velocity, ft/h

f = Fanning friction factor

g = Newton's constant = 4.1697504×10^8 ft-lbm/(lbf-h^2) .

The values of "υ" and "f" are calculated by:

$\upsilon = \dot{m}/\rho A$

$f = 16/Re$
for ($Re \leq 2000$) laminar flow

$f = 0.2086–0.187(Re/1{,}000) + 0.0624(Re/1{,}000)^2–0.0066(Re/1{,}000)^3$
for ($2{,}000 < Re < 4{,}000$) trans flow

$f = 0.079\ Re^{-0.25}$
for ($Re \geq 4{,}000$) turbulent flow ,

where,

A = flow cross-sectional area, ft^2

$\dot{m}$ = mass flow rate, lbm/h.

The Reynolds number, Re, is given by,

$$Re = \frac{4\dot{m}}{\pi\mu D_{eq}},$$

where,

μ = fluid viscosity, lbm/ft-h.

The equivalent diameter is calculated according to:

$$D_{eq} = 4\,\frac{\text{cross-sectional area of flow}}{\text{wetted perimeter}}.$$

For a circular pipe, D_{eq} is equal to the inside diameter of the pipe. For a rectangular duct, D_{eq} is given by:

$$D_{eq} = \frac{4wh}{2(w+h)},$$

where,

w = width of duct, ft

h = height of duct, ft.

The equivalent length must account for the frictional loss inside the pipe lines as well as the losses that occur due to the presence of valves, elbows, and any other fittings that involve a

change in the direction of flow or in the size of the flow passage. The head losses resulting from such fittings are functions of the geometry of the fitting, the Reynolds number, and the roughness. The method of determining the head loss in fittings is to introduce an equivalent length of pipe that produces the same head loss. Therefore, the total head loss for the piping system may be determined by adding the equivalent lengths for the fittings to the pipe length to obtain the total effective length of pipe. Typical values of L_{eq}/D for various pipe fittings are widely available in the engineering literature. Using these values, the equivalent length is easily calculated by the following equation:

$$L_{eq} = D_{eq}(L/D)_{\text{fitting}} \ .$$

The user may optionally specify each connection's equivalent length and diameter when the CoNnect command is issued in the build-up of the system layout (`CN;Leq;Deq`). The length, L_{eq}, is entered in feet and the diameter, D_{eq}, is entered in hundredths of an inch (100 = 1 inch). For instance, a connection which is 25 feet long and 6 inches in diameter should be entered as:

```
CN;25;600
```

Both values must be entered only as integers since the internal storage array may contain only integers. If no arguments are included for the `CN` command, the system automatically sets the equivalent length and diameter to their default values of 1 foot and 6 inches, respectively.

The user may select an option in which the pressure drop is given by a resistance coefficient:

$$\Delta P = K\dot{m}^2 \ ,$$

where,

K = the resistance coefficient.

The equivalent length must be entered as a negative value as follows:

$$L_{eq} = -1.0\times10^{10} * K \ ,$$

where,

K = the resistance coefficient.

For instance, a K value of 5.0E-6 is entered as:

```
CN;-50000
```

Since the connection data must be integers in the range of 1 to 99,999, a connection coefficient range of $1.0E^{-10}$ to $9.9999E^{-5}$ is possible. This option is available for the component internal flow loops by entering a negative value for L_{eq} in the data bases in a similar manner.

8.2.5 Component Internal Loop Pressure Drops

The component routines are responsible for the calculation of the pressure losses for any internal flow paths. The system level routines in turn use these values to establish the feedback loops when required. The routine PIPEDP is called by most of the components that have simple flow-through fluid paths. It calls the routine FRICTDP, described in the previous section, with the appropriate equivalent length and diameter that are usually contained in the component performance data in the CON array.

The routine COMPDP is called by those components that make specialized pressure drop calculations but also need to establish feedback loops with the rest of the system. This routine accepts the internally calculated pressure drop as an argument and sets the component stream pressures accordingly. For example, the FILTER routine must adjust the pressure drop across the filter as it gets clogged with debris and this calculation is specific only to the filtration process. However, a feedback loop is still required with the rest of the system to obtain the correct pressure balance. Therefore, the FILTER routine calculates the pressure drop across the filter element and then calls the COMPDP routine to carry out the feedback function.

The routines that simulate the components that redirect or split the flow contain specialized logic to carry out these functions. They are responsible for their own interface operations as well as maintaining the feedback loops. An example of this type is the NODE component routine, the most complicated routine from a hydraulic standpoint since it must determine the flow directions, as well as the flow rates of a variable number of connecting streams. It relies on the connecting stream pressures, connection flow path resistances, and connecting stream classifications to arrive at the correct flow balance.

The components which have "specified outlet" streams call the routine PSPEC to set their outlet pressures or establish feedback loops as appropriate. This routine is used for those outlet streams that are regulated to a set pressure or that correspond to the outlet side of an internal positive displacement flow device (constant flow rate with variable pressure differential). A nonzero value for the pressure establishes the stream as a regulated pressure stream, while a value of zero establishes a feedback loop to match the stream's pressure to the downstream conditions.

8.3 Heat Transfer Networks

The thermal network simulation techniques used in CASE/A have been developed from SINDA techniques utilized on the *Spacelab* program. These techniques include the noncondensing heat exchanger referred to as the four-pole network and the "delta" network. A description of the generic thermal network analyzer routine which is used to simulate the delta networks is also provided. Many of the discussions reference arithmetic, diffusion, and boundary nodes. These are concepts derived from SINDA. Diffusion nodes have mass (capacitance) and thus store energy. Arithmetic nodes have zero capacitance. Boundary nodes are fixed temperatures with no capacitance and do not receive impressed heat sources.

8.3.1 Noncondensing Heat Exchanger (Four-Pole Network)

Several components provide an alternate technique to the NTU method for determining a heat exchanger (HX) performance. The NTU method provides a great deal of flexibility in addressing different HX configurations (i.e., counter flow, cross-flow, etc.). The four-pole network provides an efficient computational approach to solving counterflow heat exchanges and is used in the electrolysis routines. The network representing a noncondensing HX is presented in figure 44. The values of the conductances which connect the fluid nodes may be derived from the theoretical relations for counterflow heat exchange. HX effectivity is defined as:

$$\varepsilon = \frac{1-e^{-UA/c\min(1-c\min/c\max)}}{1-(c\min/c\max)e^{-UA/c\min(1-c\min/c\max)}} \quad (1)$$

where,

ε = HX effectivity

UA = overall conductance times the area for the HX

$c$min = minimum fluid mass flow rate times the specific heat

$c$max = maximum fluid mass flow rate times the specific heat.

The HX effectivity may be used to calculate the outlet temperatures of the HX using the relations:

$$T_{o1} = T_{i1} + \varepsilon(T_{i1} - T_{i2}), \quad (2)$$

and,

$$T_{o2} = T_{i2} - (c\min/c\max)(T_{i1} - T_{i2}), \quad (3)$$

where,

T = fluid temperature

o = outlet condition

i = inlet condition

1 = side with $c$min

2 = side with $c$max.

The outlet temperatures may be related algebraically to the inlet temperatures as shown in the following equations:

$$T_{o1} = \frac{G_a\, Ti_1 + G_b\, Ti_2}{G_a + G_b}, \quad (4)$$

$$T_{o2} = \frac{G_c\, Ti_2 + G_d\, T_{i1}}{G_c + G_d}, \quad (5)$$

where,

G = constants (analogous to SINDA conductors).

The magnitudes of the energy flows within the four-pole HX network must be consistent with the rest of the component model to prevent a possible mathematical creation of energy. The

energy transfer across the HX may be written from a simple energy balance as:

$$q = \dot{m}\, C_p\, (T_o - T_i)\,. \tag{6}$$

Recognizing that the constants chosen to complete the four-pole HX network must be consistent with the above equation requires the constants G_a and G_c to be:

$$G_a = \dot{m}\, C_{p1}\,, \tag{7}$$

and

$$G_c = \dot{m}\, C_{p2}\,, \tag{8}$$

The constants G_b and G_d may be derived by equating equations (5) and (2), and equations (5) and (3) and solving for G_b and G_d, respectively. With G_a and G_c defined as in equations (7) and (8), G_b and G_d may be written as:

$$G_b = \dot{m}\, C_{p1}\, \frac{\varepsilon}{1-\varepsilon}\,, \tag{9}$$

$$G_d = \dot{m}\, C_{p1}\, \frac{\varepsilon}{1-\varepsilon\, \dfrac{\dot{m}\, C_{p1}}{\dot{m}\, C_{p2}}}\,. \tag{10}$$

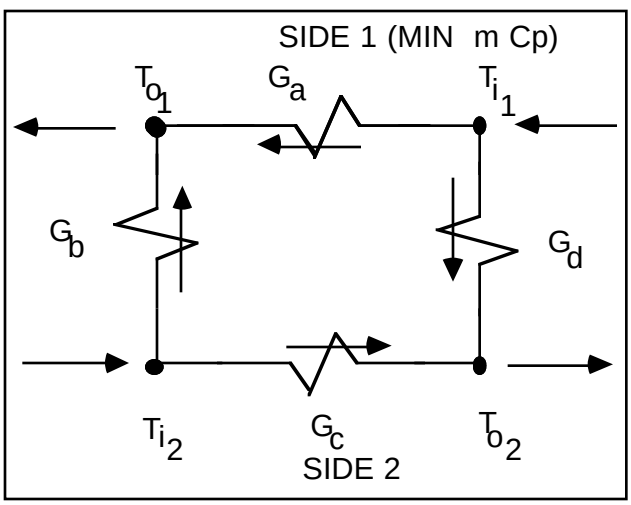


Figure 44. Four-pole noncondensing HX (SINDA analogy).

8.3.2 Delta Network

In the mathematical simulation of active ECLS components it is frequently necessary to model the heat transfer from a flowing fluid to an isothermal wall segment. This type of heat transfer process occurs in the modeling of cold plates, thermal capacitors, and the heat transfer from the rack air to equipment mass in a draw-through cooling scheme. A three-pole approach, similar to the four-pole noncondensing HX technique, is used to define this energy exchange process. The network representation of this process is shown in figure 45.

A heat exchange effectivity may be defined as:

$$\varepsilon = 1 - \varepsilon^{-UA/\dot{m}\, C_p}\,, \tag{11}$$

where,

ε = heat exchange effectivity

UA = overall conductance x area from the fluid wall segment

$\dot{m}\, C_p$ = fluid mass flow rate x specific heat.

Using this effectivity relation, the outlet fluid temperature may be expressed as:

$$T_{fo} = T_{fi} + \varepsilon (T_w - T_{fi})\,, \tag{12}$$

where,

T_{fo} = fluid outlet temperature

T_{fi} = fluid inlet temperature

T_w = wall segment temperature.

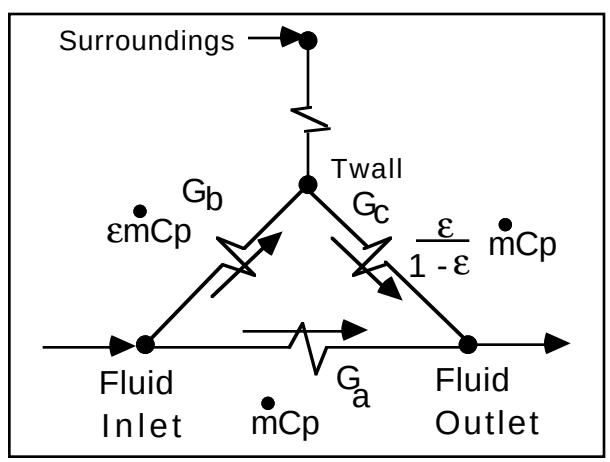


Figure 45. Delta network (SINDA analogy).

These relations may be combined to provide the outlet temperature as:

$$T_{fo} = \frac{G_a T_{fi} + G_c T_w}{G_a + G_c}, \qquad (13)$$

where,

G = constants (analogous to SINDA conductors).

From an energy balance around the fluid, the net heat addition or loss is given as:

$$q = \dot{m}\, C_p\, (T_{fo} - T_{fi}). \qquad (14)$$

Since the constants chosen to complete this three-pole "delta" network must be compatible with the above equation, the constant G_a is given as:

$$G_a = \dot{m}\, C_p. \qquad (15)$$

The constant G_c may be derived by equating equations (13) and (12) and solving for G_c with G_a as defined above. G_c is found to be:

$$G_c = \frac{\varepsilon}{1-\varepsilon}\, \dot{m}\, C_p. \qquad (16)$$

The energy transferred to the wall must represent the energy exchange with the fluid to permit the wall node to interface in a realistic manner with nodes external to the delta network.

Equations (14) and (16) may be combined to yield the relation:

$$q = \varepsilon\, \dot{m}\, c_p\, (T_w - T_{fi}). \qquad (17)$$

Constant G_b is, therefore, found as:

$$G_b = \varepsilon\, \dot{m}\, C_p. \qquad (18)$$

The three-pole "delta" network presented above explicitly provides the outlet fluid temperature and is fully equivalent to a log mean temperature difference analysis of the heat transfer process.

8.3.3 Generic Thermal Network Solution Method

Several component routines rely on a "generic" solution routine to solve their standardized thermal networks. The routine "TNETWK" provides a transient solution for the thermal network shown in figure 46 using a modified Crank-Nicholson implicit forward-backward differencing method. The basic diffusion node equation has the following form:

$$\frac{C_i}{\Delta t}(T\acute{i} - Ti) = \sum_{j=1}^{n} G_{ij} T_j - \sum_{j=1}^{n} G_{ij}\{Ti + f(\tau)[T\acute{i} + Ti]\} + Q_i,$$

where,

n = the number of nodes connected to the mass node

i = the mass node

j = the attached node

C_i = the mass node capacitance

Δt = the time interval

$T\acute{i}$ = the mass temperature after Δt

Ti = the initial mass temperature

$\tau = \dfrac{C_i}{\sum_{j=1}^{n} G_{ij}}$ = mass node time constant

Q_i = the heat load applied to the mass node

$f(\tau)$ = the first-order integration factor

G_{ij} = the linear conductor between nodes i and j.

Note the first-order integration factor

$$f(\tau) = \frac{\Delta t - \tau(1 - \exp(-\Delta t/\tau))}{\Delta t(1 - \exp(-\Delta t/\tau))},$$

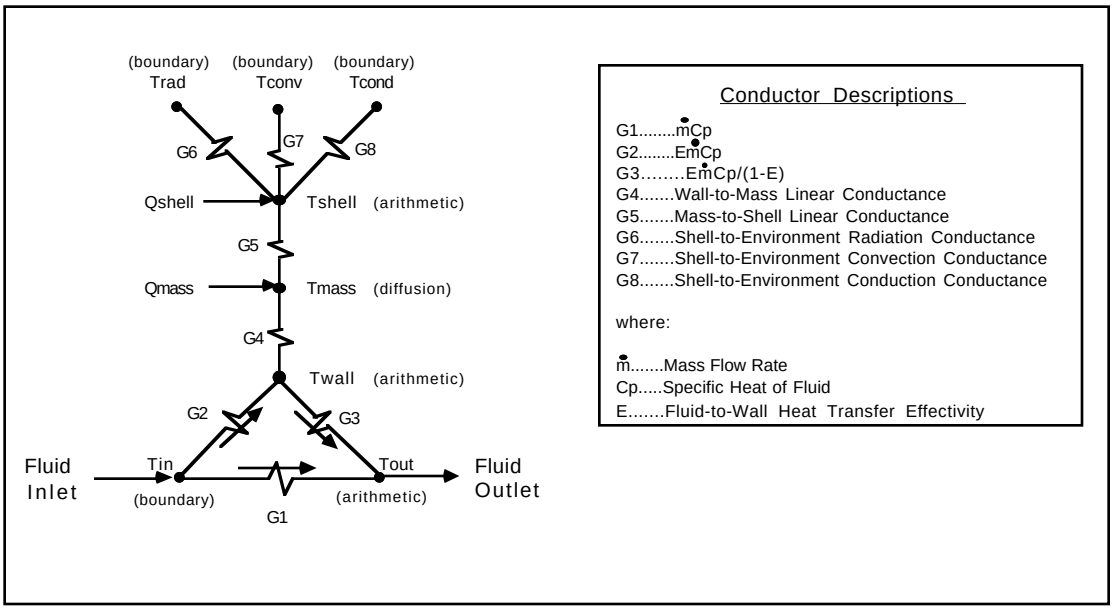


Figure 46. Thermal network solved in TNETWK.

has been used to obtain an integrated average mass node temperature over the time step for simple first-order systems.

The routine calculates the time constant for the mass node and compares this value to the specified time step passed as an argument. If the time step is greater than the time constant, the routine subdivides the time step into multiple mini-time steps, each of which corresponds to a time interval equal to the time constant. The number of minitime steps is limited to a maximum value of 10. This bounding condition insures that excessive computational time is avoided for cases which are essentially steady-state response systems. A value of 10 steps was chosen since an interval of 10 time constants results in the attainment of 99.995 percent of the equilibrium response of the mass node temperature (1.0–exp(–10)).

The basic arithmetic node equation has the following form:

$$\sum_{j=1}^{n} G_{ij}[T_j - T_i] + Q_i = 0 \,,$$

where,

i = the node whose temperature is sought

n = the number of attached nodes

j = the attached node

Q_i = the heat load applied to node i

G_{ij} = the linear or radiation conductance.

Note the quasi-linear approximation

$$G_{ij} = \sigma F_{ij} A_i E_i (T_i^2 + T_j^2)(T_i + T_j) \,,$$

has been used for the radiation term. The delta sub-network for the fluid-to-wall conductors is identical to that described in section 8.3.2.

A simple “successive point” iteration scheme is carried out on the network energy balance equations until all node temperature convergence checks fall within the specified relaxation criterion. The maximum number of iterative passes to make for each minitime step is specified as an argument. A special flag in the argument list is set if convergence is not attained for one or more ministeps within the maximum number of iterations. The calling component routine usually will print a warning message to the display if this convergence flag is set to 1.0.

8.4 Effective Wall Radiation

Analysis of radiation heat transfer from the interior surfaces of habitable area typically involves the use of thermal radiation analyzers (TRASYS) to evaluate the large radiation conductor data blocks that define the radiation transfer paths from each surface to all other surfaces. These data blocks are generally quite large (1,000 to 2,000 conductors) and consume a significant portion of computer core and computation time. For enclosures such as those encountered in a typical ECLS model, the bounding surfaces do not vary significantly in temperature (in an absolute scale). A good approximation to the detailed radiation network is the use of an effective wall radiation technique. Using this method, each node is allowed to radiate with a geometric view factor of 1.0 to an arithmetic radiation node that represents the average wall temperature for the region in question. The radiation conductor for each node in the enclosure is expressed as:

$$G = \frac{\sigma A}{\frac{1}{\varepsilon_s} + \frac{1}{\varepsilon_a} - 1}, \qquad (19)$$

where,

G = radiation conductor from the area in question to the effective wall radiation node

ε_s = IR emissivity of the surface

ε_a = area weighted average emissivity of the surfaces in the region

σ = Stefan-Boltzmann constant

A = area of surface.

Connecting the surfaces in the enclosure to the effective wall radiation node in this manner causes each node to lose energy if its temperature is higher than the mean wall, and to gain energy if its temperature is less than the mean wall. Since the effective wall radiation node is modeled as arithmetic, the thermal analyzer will calculate its temperature as a conductance weighted average of the surfaces connected to it. With the individual conductances defined as in equation (21), the effective wall radiation node will represent an area times the emissive power weighted average of the surfaces in the enclosure.

The accuracy of the effective radiation network was tested during the development of the Spacelab ECLS model. The radiation exchange in the habitable area of the *Spacelab* module was simulated using a detailed TRASYS-generated radiation conductor network. The results of this detailed simulation were compared with a mean radiant representation of the radiation exchange in the habitable area. In both simulations, the racks were loaded with equal energy dissipation and the module was placed in a nominal orbital heating environment. After 17 hours of simulated onorbit operation, the mean radiant wall temperatures differed by only 0.3 ˚F, and the avionics HX rejection differed by only 0.84 percent. The most dramatic difference between the two cases was the elapsed CPU time, with the TRASYS radiation network requiring 2.7 times that of the mean radiant method.

8.5 Mass Transfer Networks

The mass transfer process which occurs in packed beds can be simulated using the same "delta" network discussed in section 8.3.2. The analogies between the heat and mass transfer relationships are straight forward and are shown below.

Heat	Mass
$\dot{m}C_p$	$\dot{V}$
$\varepsilon\dot{m}C_p$	$\varepsilon\dot{V}$
$\frac{\varepsilon}{1-\varepsilon}\dot{m}C_p$	$\frac{\varepsilon}{1-\varepsilon}\dot{V}$
$\varepsilon = 1-e^{-\frac{\bar{h}A}{\dot{m}Cp}}$	$\varepsilon = 1-e^{-\frac{\bar{k}A}{\dot{v}}}$

where;

$\dot{V}$ = volumetric flow rates

$\dot{m}$ = mass flow rate

C_p = specific heat

$\bar{h}$ = heat transfer coefficient

$\bar{k}$ = mass transfer coefficient

A = surface area

ε = wall efficiencies.

The resulting network is identical to figure 45 with the above substitutions.

8.6 Fluid Properties Calculation

The solution system currently tracks six properties for each of the component streams (table 8). The properties of each fluid stream mixture are based on the mass weighted average values of each of the mixture's constituent properties at the given pressure and temperature. The properties are calculated and set in the PRO data array by the routine PROPS. The PROPS routine currently supports nine default constituents and allows 40 user-defined constituents. The user must modify the PROPS routine if accurate properties are needed for the user-defined constituents. Otherwise, they default to those of liquid water to at least provide nonzero values.

The properties calculations are based on the assumptions that no chemical reactions occur among the constituents and that the constituents remain in their default thermodynamic phase. Only single-phase constituents are supported at this time. Therefore, only two known thermodynamic properties of the fluid are necessary to determine the state of the fluid. The temperature and pressure were chosen as the logical properties to use as the thermodynamic state indicators. The rest of the properties are calculated based on tabular data, polynomial curve fits, or ideal gas law relationships for the gaseous constituents. The properties data were obtained from references 2 to 4.

The thermodynamic properties of each fluid flow path are propagated in a similar manner to the mass flows. That is, the temperature and pressure are set by the upstream side of a flow connection such that when combined with the mixture composition determine the other properties of the fluid for that endpoint of the connection. However, the properties of the flow streams are treated somewhat differently than the flow rate and composition in that each component stream has its own properties “node” associated with it.

The properties data are stored in this manner since pressure drop or heat transfer may occur along a flow connection path, although heat transfer is not currently enabled but easily implemented, making the two endpoints of a connection interdependent but distinct. The pressure drop and heat transfer calculations for the connection are carried out by specialized routines and the appropriately modified properties are copied into the downstream component's inlet stream array locations. These values are treated as set inlet boundary conditions in the solution of the downstream component that in turn sets the conditions of its outlet streams for the next downstream component.

8.7 References

1. “SINDA/SINFLO Computer Routine, Report No. 2-53002/4R-3167,” Vought Systems Division, LTV Aerospace Corp, 1975.

2. “ASHRAE Handbook 1981 Fundamentals,” ASHRAE, 1982.

3. Welty, J.R., Wicks, C.E., and Wilson, R.E.: “Fundamentals Of Momentum, Heat and Mass Transfer,” 1976.

4. Baumeister, T., Avallone, E.A, and Baumeister, T. III: “Mark’s Standard Handbook for Mechanical Engineers,” 1979.

9.0 USER TIPS

The CASE/A environment is designed to be intuitive, but the new user may make several common mistakes during the training process. Discussions of these common user errors and operational problems are contained in this section. Hydraulic network setup notes are also presented to aid the user in the construction of realistic models. Notes on the iterative convergence process and probable causes for convergence problems during the solution are presented next. Modeling suggestions to speed the solution process or decrease the model development time are presented.

9.1 Things to Avoid

A discussion of the common errors made during the model construction or solution phase is contained in this section.

9.1.1 Lower Case Command Input

The CASE/A command processor requires upper case character input. Any command entered in lower case will not be recognized by the command processor. The <CAPS LOCK> key on the input terminal should be engaged during interaction with the CASE/A program to avoid this error.

9.1.2 Duplicate Component Names

The user should avoid locating two or more components with the same name in a model since the data retrieval and modification routines utilize the component names to perform their functions. A warning message is written to the display if the user attempts to duplicate an existing component name. The only exception to this rule is the BUBBLE component, which requires paired names to perform its function.

9.1.3 Improper or Missing Connections

All component icon streams must be connected properly before issuing the `SOLVE` command. All streams must have at least one connecting stream. Only the flow summation or accumulation components such as the SUM, NODE, CABIN, and STORE allow multiple connections to their accumulation or input streams. A SUM is used as a flow combination device for the connection of multiple flow streams to the same component. A series of warning messages will be displayed during the solution process if improper connections exist.

9.1.4 Improper Fluid Composition Specification

The user should be careful when setting the initial fluid mass or mass flow rate and mass fractions of the STORE or SOURCE components. The mass fractions must be set to nonzero values that sum to 1.0 in order to specify the fluid composition. A common mistake is setting the total mass or mass flow rate but no mass fractions. The CASE/A program will display error messages if this situation occurs.

9.1.5 Solution Interruption During Model Loading

The solution process must never be interrupted via a <CONTROL-Y> key combination before the display of the message indicating that the simulation has been initiated. The model definition file is read into memory just after the `SOLVE` command is entered, and the file may be destroyed if an interruption occurs during this read operation. The case graphical information will then be lost, and the user must either rebuild the case from the beginning or attempt to recover the case from the CASENAME.BAK file.

9.1.6 Improper Terminal Device Setup or Specification

The user must ensure that the terminal device code set by the `TERM` command within CASE/A is compatible with the actual terminal in use (see section 6). For instance, a CASE/A terminal setting of 1 (Tektronics 4014 or compatible) will not operate properly on a VT100 or compatible terminal (proper setting is 2).

To properly support all terminal types, the VAX operating system terminal attributes WRAP/NOWRAP and ESCAPE/NOESCAPE must be set to NOWRAP and ESCAPE. This may be accomplished by entering the following statements in the user LOGIN.COM file or at the VAX prompt before CASE/A is executed:

```
$ SET TERM/NOWRAP
$ SET TERM/ESCAPE
```

9.1.7 Inadvertent Usage of System Variable Names in the OPS Routines

The user must ensure that system-reserved variable names are not used in the custom OPS routines (if any). Unpredictable results will occur since the solution system is controlled by these variables. See section 7.2.4 for a listing and discussion of the simulation control variables.

9.1.8 Incorrect Arguments to the Data Access Routines

The user must ensure that the variables that are passed to the data access routines from within the OPS routines are either declared as real or automatically default to the real type based on the variable's first character. Otherwise, either a FORTRAN run time error will occur or corrupted data will be used.

9.1.9 Incorrect Solve Request

The user must ensure that a solution request from within a version of CASE/A that contains operations logic is issued for only the model to which it applies. If a model is solved with inappropriate operations logic, the data access routines will be unable to find the missing referenced components and will write warning messages to the display device. However, the graphics and data base functions of the CASE/A environment will still operate correctly on any model.

9.1.10 Incorrect VAX Default Directory

The user must ensure that the default VAX subdirectory from which CASE/A is executed contains the desired model definition and CASEA.DAT files. Otherwise, the user will receive error messages indicating that these files are not present. See the installation notes in appendix C for further explanation of the CASE/A file architecture.

9.2 Hydraulic Solution Notes

The user must ensure that a CASE/A model is laid out correctly from a hydraulic flow standpoint. Obvious errors such as two inlets or two outlets connected together will be detected by the CASE/A program. However, the user may inadvertently create models in which there are no flow generation devices or in which there are conflicting boundary pressure nodes. Active flow generators must be present in order to drive the mass flows throughout the networks in the model. Caution must be exercised to avoid construction of unrealistic flow networks with conflicting boundary nodes. Incorrect boundary node setup will result in nonconvergence of the solution process. Discussions of these topics are presented in this section.

9.2.1 Correct Placement of Active Flow Generators

CASE/A component streams are classified as either active or passive devices relative to flow generation. Active streams will initiate fluid flow to or from the connecting components. For example, the PUMP component pulls flow from the upstream component into its inlet stream and sends the fluid through its outlet stream to the downstream component. The PUMP inlet is classified as an active inlet stream that acts as a flow sink to the upstream component. The PUMP outlet is classified as a specified outlet stream which acts as a flow source to the downstream component. The fluid flow rate versus pressure drop characteristics of the PUMP are based on the performance parameters specified by the user (see section 10.34).

Passive device streams accept the flow rates from the connecting components and act accordingly using the sensed flow rates. For example, the PIPE component consists of a passive inlet stream connected by an internal flow conduit to the passive outlet stream. Flow may be forced into the inlet side by a specified outlet stream located upstream of the PIPE or pulled from the outlet side by an active inlet stream located downstream of the PIPE (fig. 47). In either case, the flow will be transferred from the inlet side to the outlet side with an internal flow conductor established between them.

Flow may be pulled or pushed through passive components. Note on figure 47 that the stream fill patterns on the component icons indicate their hydraulic stream type (see section 8.2.3).

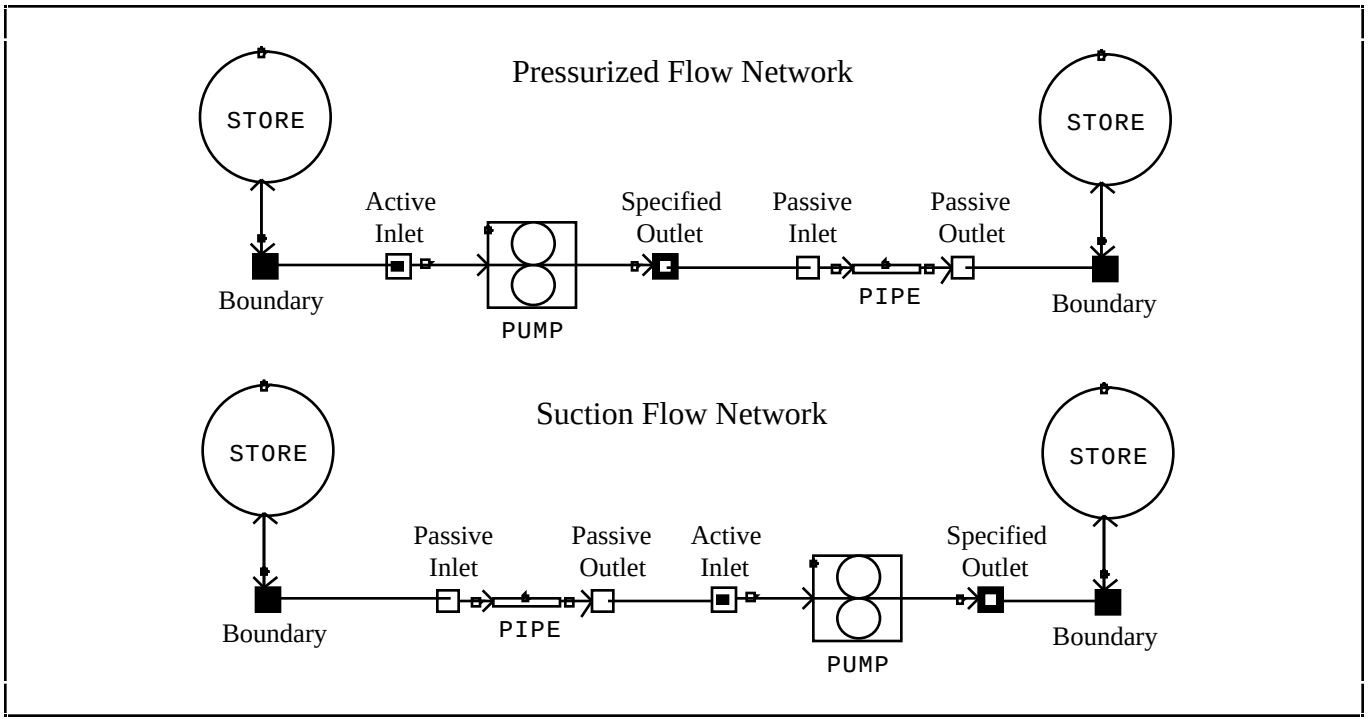


Figure 47. Example flow generator placement schematics.

9.2.2 Placement of Boundary Pressures

CASE/A component streams are classified as either variable, boundary, or stagnant pressure nodes. Their classifications may change during a simulation based on the operational status of the components. For instance, the streams of a PUMP become stagnant if it is deactivated and those of the VALVE also become stagnant if it is closed. Variable and stagnant pressure nodes adjust to the flow configuration and the boundary pressures in the network. All passive inlet or outlet streams are variable pressure nodes and may become stagnant nodes.

All specified outlet streams (flow sources) may be designated as either boundary or variable pressure nodes. They are treated as boundary nodes if their pressures are set in the component performance parameters to nonzero values. However, they are treated as variable pressure nodes if their pressures are set to 0.0 (see section 8.2.3).

Each flow subnetwork should contain at least one stream that acts as a boundary node. Two boundary nodes at opposite ends of the same flow branch should be avoided (fig. 48) and will cause nonconvergence.

9.3 Solution Convergence Notes

The system convergence criterion and maximum iteration loops are specified in the CONTROL file and should be set with care. The maximum number of iteration loops defaults to 75, and the convergence criterion defaults to 1.0E-4. If the convergence criterion is very small (on the order of 1.0E-6), the number of iteration loops required to attain convergence will increase dramatically for complex or highly transient models.

PLEASE NOTE: If the user finds that the model is not converging for cases with a large number of components, please try the “Feedback” solution method (specified in the CONTROL file). In the past, problems with convergence have been noted where the “Matrix” solution method was used for large cases.

9.3.1 Convergence Criterion Versus Iteration Loops

The maximum number of iteration loops and the system convergence criterion are closely interrelated (see section 8.1). The system will only perform the number of iteration loops required to achieve the specified convergence criterion. A

decrease in the convergence criterion will usually necessitate an increase in the maximum number of loops. A large number of system loops may be required for startup transients. Complex model simulations will require larger loop counts and a corresponding increase in computational time.

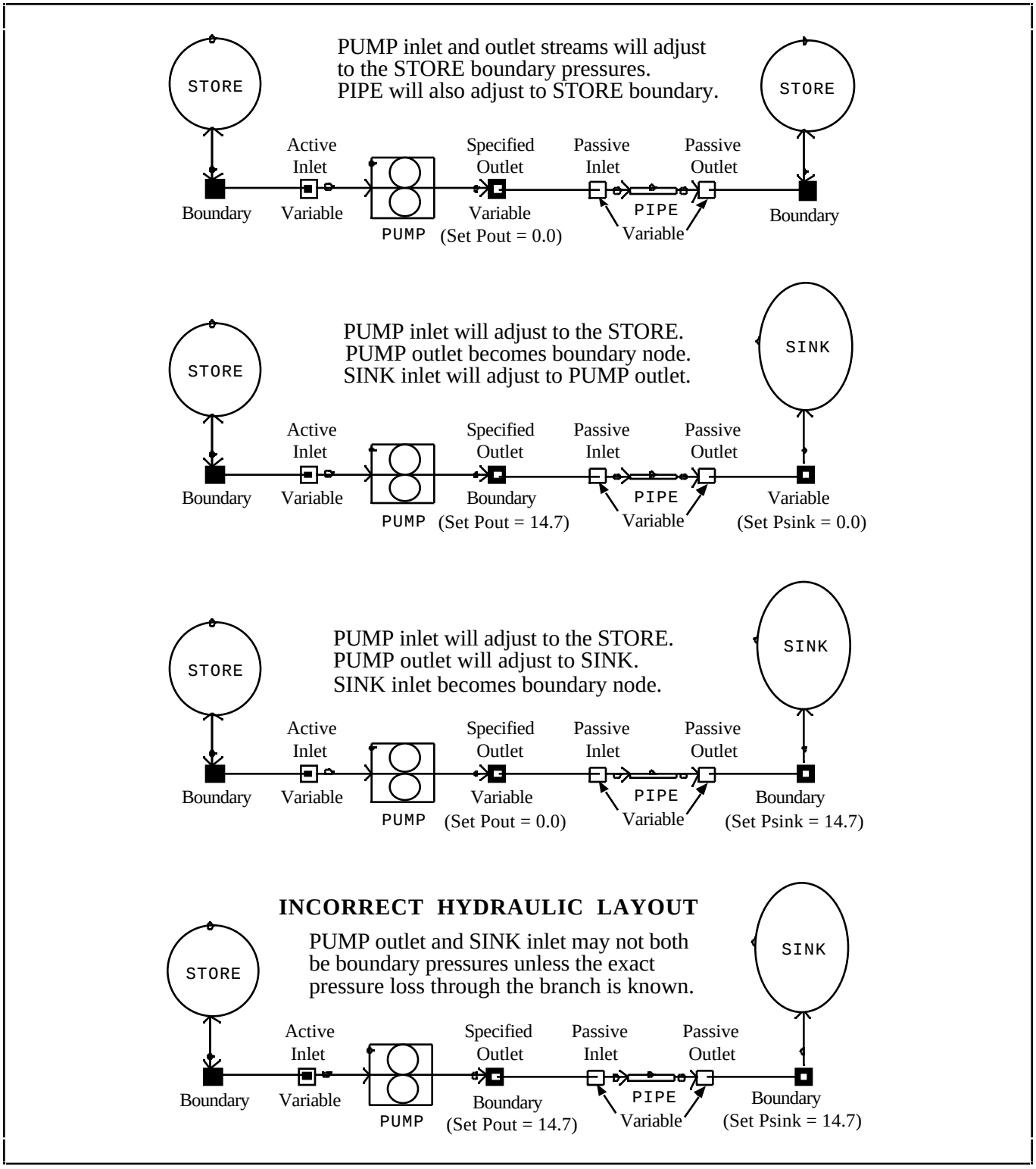


Figure 48. Example boundary pressure placement schematics.

Experimentation on an individual model basis is required to attain an optimum balance between degree of convergence and maximum number of iterations. If the system does not meet the convergence criterion during a time step, a list of the components which did not converge and their convergence indication values is written to the display device. If the convergence indication values are close to the desired criterion, the user may wish to continue the simulation with these larger numerical errors.

9.3.2 Probable Causes for Nonconvergence

A convergence indication value of 1.0 for any component stream indicates a problem with either the case configuration, operating parameters, or the CASE/A solution routines. The case configuration should first be checked for obvious connection errors based on the component streams that are not converging. Also, the operating parameters of the problem components should be double checked. The user should be aware that a single nonconvergent stream may be responsible for propagating the error to all of the components in the flow path. Therefore, the investigation of the problem should begin with the first component in the flow path. Incompatible boundary pressure streams will usually be indicated by nonconvergence of the pressure solution. The user should refer to section 8.2 for further information on the pressure balance solution.

Typically the number of iteration loops may be high for startup transients when initial system conditions are not at their equilibrium values. The system may not reach convergence in the first few time steps. However, it may eventually attain convergence after a few time steps have been completed and more stable conditions have been reached. This situation is indicated by a steady decreasing of the convergence indication values with each passing time step. The maximum number of iterations should be increased progressively by multiples of two to see if the system will initially converge. If it does not converge even with a large number of loops (on the order of 500 loops), something may be wrong with the input performance parameters, case configuration, or with the solution routines themselves.

9.4 Modeling Suggestions

This section discusses modeling tips that speed the model development process or the solution process. The most important of these tips is the construction of large cases in a modular fashion. The usage of SOURCE and SINK components instead of PUMP and STORE components is discussed. Usage of the CABIN component as a common thermal environment for a group of components is discussed next. Finally, a discussion of the transient thermal calculations is presented.

9.4.1 Modular CASE Construction

It is recommended that large CASE/A models be constructed in a modular fashion to speed the development process. This may be accomplished by constructing large subsystems as separate models and then merging them into the overall model with the `MERGE` command. This approach allows more rapid debugging of model or OPS logic problems due to the faster turn around time of the smaller case. Also, the development effort may be carried out by several different users in their own subdirectories or even on different VAX host computers.

9.4.2 SOURCES and SINKS Speed Solution Process

The user may elect to use the SOURCE and SINK components instead of the PUMP and STORE components as flow generators or boundary nodes to decrease the solution execution time. The PUMP and STORE components provide much more fidelity in their mass, pressure, and thermal simulation calculations than do the SOURCE and SINK. Therefore, their computational requirements are much higher. A SOURCE and SINK combination may be used as a replacement for a PUMP and STORE combination when the user needs certain boundary flow conditions but no detailed pressure balancing, mass accumulation, or heat exchange characteristics.

9.4.3 Usage of a Cabin as a Thermal Environment

The CASE/A program allows the user to link the component heat loads to the CABIN

components to simulate heat exchange with the environment. This is accomplished via the `ASSIGN` command which allows the user to assign all of the components in a given subsystem to a specified CABIN in the case. The CABIN may be used as a fixed thermal boundary node for a group of components by setting the structure mass to a very large value and setting the CABIN initial contents and sink temperatures to the desired values (see section 10.8). If a subsystem is not assigned to a CABIN, the default thermal environment is set at 75 ˚F.

9.4.4 Thermal Steady-State Versus Transient Calculations

The component energy balance calculations may be transient or steady-state as dictated by the component type and user-specified thermal capacitances. Those components that allow the specification of thermal capacitance support transient thermal calculations on a quasi steady-state basis (see section 8.3). The default capacitance values depend on the equipment type. The user may set the capacitances to 0.0 in the component data bases to activate the steady-state thermal network solutions. Transient calculations are automatically performed when no-zero values are specified.

9.4.5 Usage of a Valve to Generate Fluid Flow

The VALVE component supports an option in which it acts as a flow generation device. The flow rate and direction are based on the boundary pressures and flow configuration of the network in which it is located. It may be utilized with the STORE component to simulate a pressurized tank that drives flow through the downstream network. This option is toggled via the 28th input location of the VALVE data base (see section 10.49).

10.0 COMPONENT LIBRARY DESCRIPTION

An overview of the functions, input/output parameters, and modeling techniques and limitations of each component routine is provided in this section. Table 17 shows a complete list of ECLSS/ATCS components included with CASE/A version 4.0. The component descriptions are arranged alphabetically.

Each ECLSS and ATCS component is represented by a unique geometric figure (icon) that the user can manipulate by the commands documented in section 2. Component descriptions provide an overview of each component, stream definitions of the icon, and input data needed from the user. The input data can be submitted or modified by the `ED` or `E` command as discussed in section 3.0.

Components can be classified as either active or passive devices relative to flow generation. Active devices will generate a flow rate; for example, a pump that generates flow rates according to the rate specified by the engineer. Passive devices accept the flow rates from the upstream component and operate appropriately on that flow rate. The output of a passive device is a function of the input flow rate plus or minus the internal losses or redirection of flow. The output of an active device is specified by the user. In some cases a component can be a mixture of both. For example, the BOSCH routine, which simulates the Bosch CO_2 reduction process, accepts concentrated carbon dioxide from an upstream component that may or may not be mixed with the required quantity of hydrogen. If the incoming mixture is deficient in hydrogen, the component will attempt to draw the required hydrogen from stream number 1, which should be connected to a hydrogen tank in accordance with the component design. In general, air streams from a cabin must be pulled through a pump/fan component to generate a flow rate. See section 8.2 for a more detailed discussion of the pressure drop solution methodology.

The component routines use concepts and techniques derived from both SINDA and G189A. Routine descriptions assume that the reader is familiar with SINDA and the R-C analogy used to simulate heat transfer. These concepts and modeling techniques are discussed in section 8.0 of this document. References to the G189A and SINDA manuals are noted in the component descriptions where appropriate.

The user can modify component subroutines and/or add subroutines to the system. This will require that new routines be compiled and linked to the CASE/A library. This must be done externally to the CASE/A program. The user must then create a new executable and request a solution. Command files are provided to aid the user in generating new executables.

The input parameters deserve additional discussion since they are a key element to the proper simulation of the component. The item number for references from OPS blocks (see section 7.0 on OPS blocks) are included for all input parameters. The screens shown are those the user would see when using a terminal that emulates a DEC VT100™ terminal. This is the preferred approach to editing the component data parameters. An alternate line editor is provided for those users who do not have access to the newer advanced graphics terminals that emulate both the Tektronix graphics and the DEC VT100™. In some cases, a component contains more data parameters than would fit on a single screen. Multiple screens are provided for these components. The user can switch to the next screen by toggling to the appropriate location and entering a `<Ctrl L>`. This opens the next screen for user review and/or modification. The user enters a `NEXT SCREEN` or `<Ctrl Z>` to save the data and return to the previous level. This function will work only on those devices which provide a VT100™ emulation. Otherwise, the edit sequence involves listing the parameters and modifying by specifying the item number to be changed and the new value (see section 3.3).

Table 17. Available equipment types.

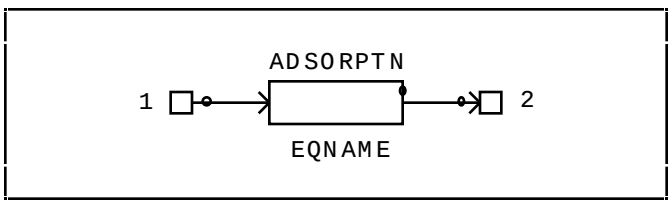
EQTYPE MNEMONIC	COMPONENT SIMULATED
ADSORPTN	Adsorption Bed
AFSPE	Anode Feed Solid Polymer Electrolysis
ALHX	Air/Liquid Heat Exchanger
BLACKBOX	User-Defined Component
BMR	Body Mounted Radiator
BOSCH	Bosch Reactor System
BUBBLE	System Interconnect Bubble
CABIN	Cabin
CAP	Thermal Capacitor
CFR	Carbon Formation Reactor
CHX	Condensing Heat Exchanger
CNHX	Contact Heat Exchanger
CNTRLLR	Controller
CP	Cold Plate
CREW	Crew Metabolic Parameters
DEFLOW	Dehumidifier-Flow Desorbed
DEHUM	Dehumidifier-Vacuum Desorbed
EDC	Electrochemical Depolarized Concentrator
EVAP	Flash Evaporator
FILTER	Filter Device
H2OSEP	Water Separator
HATCH	Hatch/Tunnel
HEATER	Flow-Through Heater
IONEXCH	Ion Exchange Resin Unit
LIOH	Lithium Hydroxide Carbon Dioxide Removal System
LLHX	Liquid/Liquid Heat Exchanger
MODULE	Module
MOLSIEV	Molecular Sieve for Carbon Dioxide

Table 17. Available equipment types (continued).

EQTYPE MNEMONIC	COMPONENT SIMULATED
MSPLIT	Flow Multi-Split Device
NODE	Pressure/Flow Manifold
O2N2	Oxygen/Nitrogen Gas Control Panel
PIPE	Pipe Section
PREWAST	Waste Processing Pretreatment
PUMP	Fluid Transfer Pump
RACK	Cabin Electronic Equipment Rack
RAD	Single-Phase Radiator
RO	Reverse Osmosis Unit
SABAT	Sabatier Reactor System
SAWD	Solid Amine Water Desorbed
SFWE	Static Feedwater Electrolysis
SINK	Ideal Fluid Sink
SOURCE	Ideal Fluid Source
SPLIT	Flow Stream Splitter
STORE	General Storage Tank
SUM	Flow Stream Combination
TBUS	Thermal Bus
TIMER	Timer
TIMES	Thermoelectric Integrated Membrane Evaporation System
VALVE	Fluid Valve
VCD	Vapor Compression Distillation
WASH	Generalized Wash Process
WQM	Water Quality Monitor

10.1 ADSORPTN—Adsorption Bed Simulation Description

The ADSORPTN routine will predict the performance of a fixed cylindrical adsorption bed containing carbon or any other appropriate material as the adsorber. The user provides the performance characteristics for a given adsorber material and bed length from which the routine will predict the performance and operational life of the adsorption bed. The removed solids accumulate in the adsorber material until a user-defined breakthrough concentration for any constituent is exceeded, at which point the bed is changed out. The total number of beds required during the simulation is recorded. The ADSORPTN component is a "passive" device with respect to mass flow, i.e., it only accepts the upstream mass flow or relies on a downstream component to pull flow through it. Therefore, an "active" flow-generating component such as the PUMP must be located upstream or downstream.



10.1.1 General Assumptions

The development of this routine was based upon a number of simplifying assumptions. The first assumption requires that there be no axial dispersion in the adsorber. The next restricts the method to those systems for which the equilibrium isotherm is linear. The third requires that a proportionality exist between the adsorption rate, liquid solute concentration, and residual material capacity. The fourth assumes a functional relationship between diminishing rate of the solute concentration across the bed and the equilibrium isotherm. Finally a symmetrical S-shaped breakthrough curve is assumed. The method of solution is known as the bed-depth/service-time (BDST) analysis. The operating equations and simplifying assumptions were obtained or developed from references 1 and 2.

10.1.2 Mass and Pressure Calculations

The user provides three configuration variables, two adsorption material constants, four heat transfer coefficients, and three coefficients for each dissolved solute in the feed stream. Each input is discussed in detail below.

The configuration variables consist of the diameter of the adsorption column, the length of the adsorption column, and the flow resistance constant per unit length of the adsorption column given by the following equation:

$$R = (0.7854 g_c D^2 \Delta P)/(\mu Q L) ,$$

where,

D = adsorption column diameter (ft)

g_c = Newton's constant = 4.1697504×10^8ft•lbm/(lbf•h^2)

L = adsorption column length (ft)

ΔP = pressure drop across the column (lbf/ft^2)

Q = volumetric flow rate (ft^3/h)

μ = feed stream viscosity (lbm/ft•h).

The flow resistance need only be evaluated once for a given adsorption material and column size. The routine will then calculate the pressure drop for any size column, assuming that the porosity of the adsorption material remains the same for different column sizes, by the equation above.

The two adsorption material constants consist of the solids holding capacity of the material and the adsorption constant of the material. These constants are evaluated for a single reference solute by the following equation:

$$t = \frac{0.7854\, D^2 No\, Z}{C_i Q} - \frac{1}{K C_i} \ln\left[\frac{C_i}{C_o} - 1\right],$$

where,

t = elapsed time (h)

Z = bed length (ft)

D = bed diameter (ft)

Q = volumetric flow rate (ft^3/h)

C_i = inlet solute concentration (lbm/ft^3)

C_o = outlet solute concentration (lbm/ft^3)

N_o = material capacity (lbm/ft^3)

K = adsorption constant (ft ^{3}/lbm•h).

The volumetric flow rate and the inlet solute concentration should be held constant for this equation to be applicable. The solution of N_o and K requires the measurement of the outlet concentration at two different times and solving two simultaneous equations for N_o and K.

The user provides three coefficients for each solute in the feed stream. Two of the coefficients account for the variation of the adsorption constant and capacity from the reference solute values and are calculated by the following equation:

$$t = \frac{0.7854\, D^2 N_o\, Z}{C_i Q}\, a - \frac{b}{K C_i}\, \ell n \left[\frac{C_i}{C_o} - 1 \right],$$

where,

a,b = constants for the given solute

N_o = reference solute material capacity

K = reference solute adsorption constant.

The solution for a and b requires the measurement of the inlet and outlet concentrations of the given solute at two different times from which two simultaneous equations may be solved for a and b. These values are entered as 1.0 for the reference solute. The user may choose any reference solute in the determination of N_o and K.

The third solute coefficient is the maximum allowable concentration of the solute in the effluent stream. When this "breakthrough" concentration is reached, the routine changes out the bed and increments the number of beds required during the simulation.

10.1.3 Thermal Calculations

An R-C network analysis technique is utilized in the thermal calculations and is based on the generic network discussed in section 8.3.3 which allows for three modes of heat transfer with the environment: convection, conduction, and radiation. The total mass of the ADSORPTN column is lumped together to form the mass diffusion node. The network solution calculates the fluid outlet and structural temperatures based on the user-specified conductances, thermal capacitance of the mass node, and environment and fluid inlet boundary temperatures. There are four linear conductances the user must provide as inputs along with the emissive area term for radiation and a fluid-to-wall heat transfer effectivity. Also, the user must specify an initial mass temperature, a relaxation criterion for all temperatures and a maximum iteration count not to be exceeded in the solution routine. The solution routine also allows for heat loads to the mass and external shell nodes. In the ADSORPTN calculations, the shell node heat load is set to zero and the mass node heat load is set to zero.

10.1.4 Data Base Parameters

The ADSORPTN data base parameters are shown in table 18. The ADSORPTN constituent data base parameters are shown in table 19. The default data are set such that there is no heat exchange with the environment and no thermal capacitance (steady-state thermal solution). The constituent constants are set so that all constituents will pass directly through the column with no adsorption.

References

1. Noreen, T.R.: "Application of the Finite Compartment Model of Carbon Adsorption to Binary Solute Systems." M.S thesis, May 1984.

2. Treybal, R.E.: "Mass Transfer Operations," McGraw-Hill, 1980.

Table 18. ADSORPTN main edit screen.

```
CASE NAME: DEFAULT                                ADSORPTN EDIT SCREEN                                LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                  RECORD NUMBER: 000001

|          ** GENERAL INPUT DATA **          |    ** THERMAL CHARACTERISTICS DATA  **    |     ** MISCELLANEOUS  PARAMETERS **      |
| 10 ADSORBR COLUMN DIAMETR (FT)      0.250  | 15 SHELL-ENV CONVCT (BTU/H/F)      0.000  | 25  ** OPEN INPUT LOCATION **     0.000  |
| 11 ADSORBR COLUMN LENGTH  (FT)      1.000  | 16 SHELL-ENV RAD  FAE   (FT2)      0.000  | 26  ** OPEN INPUT LOCATION **     0.000  |
| 12 COLUMN FLOW RESIST  (1/FT2)  0.100E+11  | 17 SHELL-ENV CONDCT (BTU/H/F)      0.000  | 174 ** OPEN INPUT LOCATION **     0.000  |
| 13 TOTAL CAPACITY    (LBM/FT3)     22.640  | 18 MASS-SHEL CONDCT (BTU/H/F)      1.000  |                                          |
| 14 ADSORP CONSTANT (FT3/LBM/H)      1.620  | 19 WALL-MASS CONDCT (BTU/H/F)      1.000  |        ** RESOURCE TRACKING **           |
|                                            | 20 FLUID-WALL EFFECT  (0<X<1)      0.950  | 7  POWER     (WATTS)              0.0000 |
|                                            | 21 THERMAL CAPACITNCE (BTU/F)      0.000  | 8  WEIGHT      (LBM)              0.0000 |
|                                            | 22 INITIAL MASS TEMPRTURE (F)     75.000  | 9  VOLUME     (FT^3)              0.0000 |
|                                            | 23 THERMAL RELAXATN CRITERIA   0.0000100  |                                          |
| -- CONSTITUENT EMP DATA  (^L  TO OPEN ) _  | 24 THERMAL MAX ITERATN COUNT     100.000  |                                          |

|                        ** BENCHMARK DATA **                  |  ** OUTPUT DATA **                                          |
|                          MIN          NOM          MAX       | 177  NUMBER OF UNITS REQUIRED                    0.000    |
|  PRESSURE DROP   (PSID)      0.000        0.000        0.000 | 178  PRESENT PRESSURE DROP        (PSID)         0.000    |
|  PRODUCT  FLOW  (LBM/H)      0.000        0.000        0.000 | 179  PRESENT PRODUCT FLOW RATE   (LBM/HR)         0.000    |
|                                                               | 180  PRESENT MASS  TEMPERATURE        (F)         0.000    |
| 175  PRESENT UNIT,S SERVICE TIME         (HR)          0.000 | 181  PRESENT SHELL TEMPERATURE        (F)         0.000    |
| 176  TOTAL ACCUM SOLIDS FOR ALL UNITS   (LBM)          0.000 | 182  **  OPEN OUTPUT LOCATION  **                 0.000    |
```

Table 19. ADSORPTN constituent edit screen.

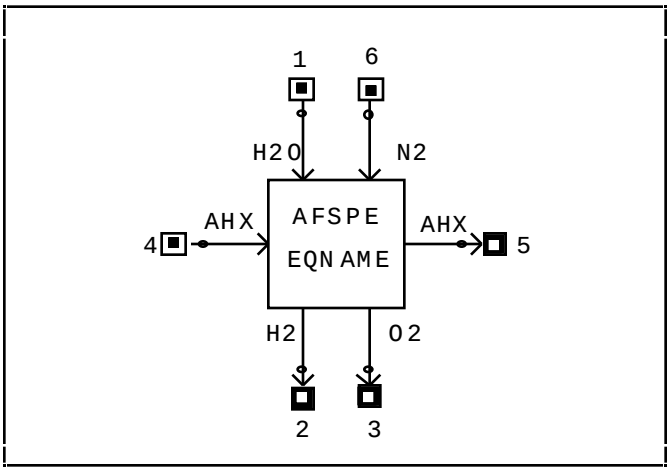
```
CASE NAME:  DEFAULT                               ADSORPTN CONSTITUENT SCREEN                          LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                  RECORD NUMBER: 000001

|   LABEL    | EMPIRICL | EMPIRICL | BRKTHRU |  LABEL   | EMPIRICL | EMPIRICL | BRKTHRU |  LABEL   | EMPIRICL | EMPIRICL | BRKTHRU |
|            | CNSTNT A | CNSTNT B |  (PPM)  |          | CNSTNT A | CNSTNT B |  (PPM)  |          | CNSTNT A | CNSTNT B |  (PPM)  |
|OXYGEN      0.000E+00 0.000E+00 2000000.0 CONSTIT 19   0.000E+00 0.000E+00 2000000.0 CONSTIT 36   0.000E+00 0.000E+00  2000000.0 |
|NITROGEN    0.000E+00 0.000E+00 2000000.0 CONSTIT 20   0.000E+00 0.000E+00 2000000.0 CONSTIT 37   0.000E+00 0.000E+00  2000000.0 |
|CARBON DIOX 0.000E+00 0.000E+00 2000000.0 CONSTIT 21   0.000E+00 0.000E+00 2000000.0 CONSTIT 38   0.000E+00 0.000E+00  2000000.0 |
|HYDROGEN    0.000E+00 0.000E+00 2000000.0 CONSTIT 22   0.000E+00 0.000E+00 2000000.0 CONSTIT 39   0.000E+00 0.000E+00  2000000.0 |
|CARBON (S)  0.000E+00 0.000E+00 2000000.0 CONSTIT 23   0.000E+00 0.000E+00 2000000.0 CONSTIT 40   0.000E+00 0.000E+00  2000000.0 |
|METHANE     0.000E+00 0.000E+00 2000000.0 CONSTIT 24   0.000E+00 0.000E+00 2000000.0 CONSTIT 41   0.000E+00 0.000E+00  2000000.0 |
|WATER       0.000E+00 0.000E+00 2000000.0 CONSTIT 25   0.000E+00 0.000E+00 2000000.0 CONSTIT 42   0.000E+00 0.000E+00  2000000.0 |
|FREON-11    0.000E+00 0.000E+00 2000000.0 CONSTIT 26   0.000E+00 0.000E+00 2000000.0 CONSTIT 43   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 10  0.000E+00 0.000E+00 2000000.0 CONSTIT 27   0.000E+00 0.000E+00 2000000.0 CONSTIT 44   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 11  0.000E+00 0.000E+00 2000000.0 CONSTIT 28   0.000E+00 0.000E+00 2000000.0 CONSTIT 45   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 12  0.000E+00 0.000E+00 2000000.0 CONSTIT 29   0.000E+00 0.000E+00 2000000.0 CONSTIT 46   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 13  0.000E+00 0.000E+00 2000000.0 CONSTIT 30   0.000E+00 0.000E+00 2000000.0 CONSTIT 47   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 14  0.000E+00 0.000E+00 2000000.0 CONSTIT 31   0.000E+00 0.000E+00 2000000.0 CONSTIT 48   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 15  0.000E+00 0.000E+00 2000000.0 CONSTIT 32   0.000E+00 0.000E+00 2000000.0 CONSTIT 49   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 16  0.000E+00 0.000E+00 2000000.0 CONSTIT 33   0.000E+00 0.000E+00 2000000.0 CONSTIT 50   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 17  0.000E+00 0.000E+00 2000000.0 CONSTIT 34   0.000E+00 0.000E+00 2000000.0                                             |
|CONSTIT 18  0.000E+00 0.000E+00 2000000.0 CONSTIT 35   0.000E+00 0.000E+00 2000000.0                                             |
```

10.2 AFSPE—Anode Feed Solid Polymer Electrolysis Description

The AFSPE routine simulates the solid polymer electrolysis method of H_2 and O_2 production through the electrolysis of water. The chemical reactions governing this process are given as follows:

Anode: $2H_2O \text{--->} O_2 + 4H^+ + 4e^-$
Cathode: $4H^+ + 4e^- \text{--->} 2H_2$
Overall: $2H_2O \text{--->} 2H_2 + O_2$



10.2.1 General Discussion

The AFSPE produces O_2 through the electrolysis of water at the anode of each cell. When water is electrolyzed at the anode, hydrogen ions and electrons are produced as well as O_2. Electrons travel through the cell external circuitry to the cathode. The hydrogen ions migrate from the anode to the cathode along with protonically pumped water where they recombine with electrons on the cathode to evolve as molecular hydrogen. Water is supplied to the anode by a continuously recirculating feed loop that also provides the thermal control for the cell stack. The O_2 produced at the anode leaves the stack with the recirculating water loop and from there enters a separator that separates O_2 gas from the water. Saturated O_2 leaves the separator while the water reenters the recirculation loop. The recirculation water then mixes with makeup water and water from the H_2 separator and travels through the air/liquid HX and deionizer before reentering the cell stack.

H_2 formed at the cathode leaves the cell stack with protonically pumped water and enters the H_2/H_2O separator. Saturated H_2 leaves the separator, while water from the separator is mixed with makeup water passed back to the recirculation loop.

Cell stack temperature is controlled by turning the fan in the air/liquid HX on when the cell stack fluid outlet temperature exceeds the upper limit of the temperature control band. The fan is turned off when the stack outlet temperature reaches the lower limit of the control band.

Inlet stream 1 is used to supply makeup water to the subsystem. Stream 4 provides air for the liquid/air HX used in the recirculation loop and stream 6 supplies N_2 purge gas to the component during the startup phase. Streams 1, 4, and 6 are active inlet streams that pull in flow from the upstream components.

10.2.2 Mass Balance

The H_2O feed rate is determined based on the H_2O required to replace the water that is electrolyzed and leaves as water vapor in the H_2 and O_2 outlet streams. This feed rate is combined with the water leaving the H_2/H_2O separator. These combined streams then resupply H_2O to the recirculating feed loop. The H_2O feed rate into the recirculating loop alternates between two flow rates depending upon the volume of the high-pressure accumulator. While in the normal operating mode, the H_2O feed rate is 6.0 lbm/h until the accumulator volume exceeds 20 in^3, at which time the feed rate is reduced to 4.5 lbm/h. When the accumulator volume drops below 10 in^3, the feed rate returns to 6.0 lbm/h. In the emergency operating mode, the H_2O feed rate cycles between 6.0 and 7.5 lbm/h. Given the resupply rate into the recirculating loop, the flow rate of the H_2O makeup stream (stream 1 on the component icon) is calculated as follows:

$$\dot{m}_{\text{makeup}} = \dot{m}_{\text{in}} - \dot{m}_{\text{prot}} + \dot{m}_{wv}$$

where,

$\dot{m}_{\text{in}}$ = H_2O feed rate into the recirculating loop $\left(\frac{\text{lbm}}{\text{h}}\right)$

$\dot{m}_{prot}$ = rate of protonically pumped water in the cell stack $\left(\frac{lbm}{h}\right)$

$\dot{m}_{wv}$ = rate of H_2O leaving as water vapor in the H_2 stream $\left(\frac{lbm}{h}\right)$.

H_2, O_2, and H_2O electrolysis rates are calculated based on the governing chemical equations and the Faradaic efficiency (FE) of the cells. The desired oxygen production rate is set by the user and the current input required to obtain this rate is determined by the following equation:

$$\text{current} = \frac{\dot{m}_{O_2}\left(4519.7\ \frac{\text{Amp}\bullet\text{h}}{\text{lb } O_2}\right)}{\text{FE}}.$$

The H_2 production and H_2O electrolysis rates are then determined based on this current value using the following equations:

$$\dot{m}_{H_2} = \text{FE}\left(\frac{\text{cell current}}{12062.0\ \frac{\text{Amp}\bullet\text{h}}{\text{lb } H_2}}\right),$$

$$\dot{m}_{H_2O} = \text{FE}\left(\frac{\text{cell current}}{1349.9\ \frac{\text{Amp}\bullet\text{h}}{\text{lb } H_2O}}\right).$$

The amount of protonically pumped water is calculated based on a user input ratio of protonically pumped water to the water electrolysis rate. This value is multiplied by the water electrolysis rate, determined as shown above, to obtain the rate of protonically pumped water in each of the cells.

10.2.3 Thermal Calculations

The thermal network used to model the AFSPE consists of a series of mass nodes corresponding to each cell and the divider plates separating the cell stack from the O_2 and H_2 water separators. Figure 49 shows the thermal network scheme used for cells 2 through n-1, where n is the number of cells in the stack. The mass node represents the lumped capacitance of a single cell assembly. The mass-to-wall conductance represents conduction from the cell to the wall adjacent to the coolant pass in the next cell. The recirculating loop mass flow rate into the cell stack is split equally among the individual cells, with the same cooling effectivity for each cell. Mass to shell node conductance and environment conductances are assumed to be the same for all cells. Figure 50 shows the thermal network used for the first cell in the stack and the divider plate next to the H_2/H_2O separator. T_{exit} corresponds to the exit temperature of the fluid from the cell stack. G_h represents the fluid flow conductance into the H_2/H_2O separator based on the rate of protonically pumped water in the stack. A similar thermal network configuration was constructed for the divider plate and cell next to the O_2/H_2O separator and is shown in figure 51. G_{ox} represents the fluid flow conductance in the O_2/H_2O separator adjacent to the divider plate. In the present configuration, the flow in the O_2/H_2O separator is split into four streams before leaving the separator, with one of these streams passing by the cell stack divider plate. G_{ox} was calculated based on 25 percent of the fluid flow leaving the cell stack, less the protonically pumped water and H_2 going to the H_2/H_2O separator. With the fluid conductance into each cell given as $\dot{m}c_p$ (G_3), G_h, and G_{ox} are defined as follows:

$$G_h = \dot{m}_{prot} x c_p$$

$$G_{ox} = 0.25(\dot{m} x c_p - G_h)\ ,$$

where,

x = number of cells in the stack

c_p = specific heat of the fluid $\left(\frac{\text{Btu}}{\text{lbm-°F}}\right)$

$\dot{m}_{prot}$ = flow rate of protonically pumped water $\left(\frac{lbm}{h}\right)$.

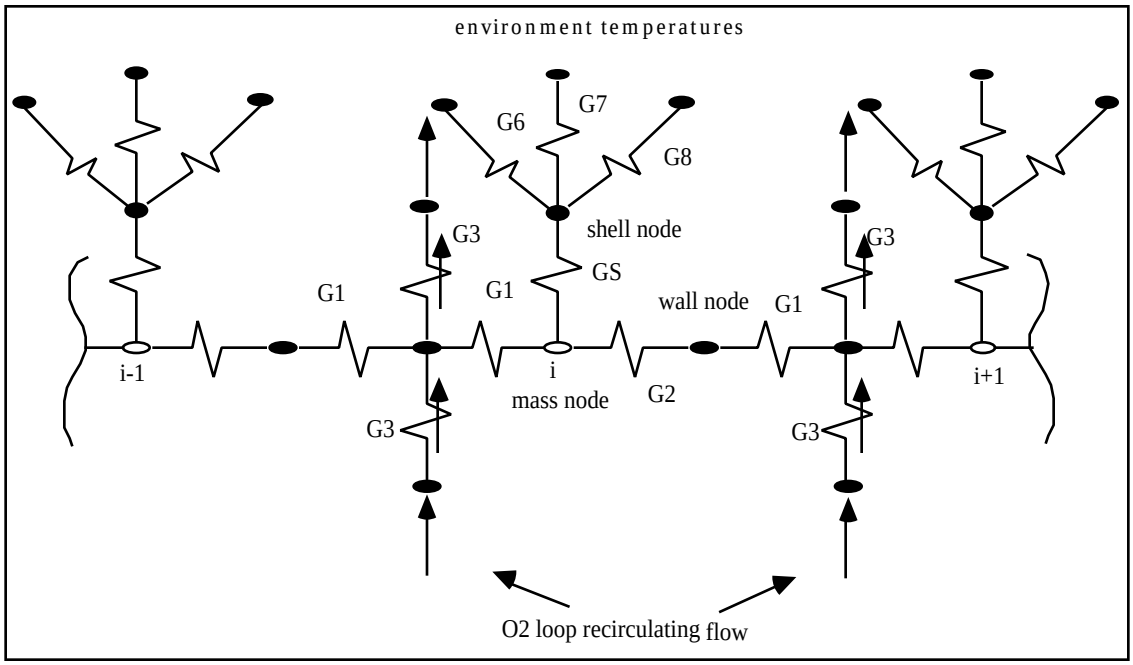


Figure 49. Single cell thermal network.

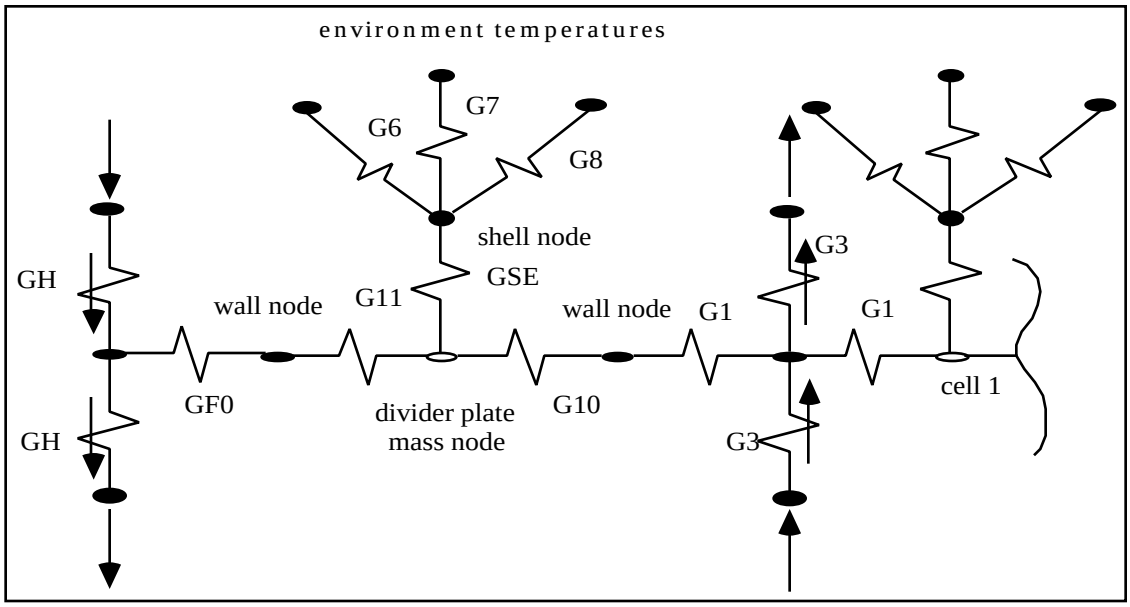


Figure 50. Divider plate node next to H_2/H_2O separator.

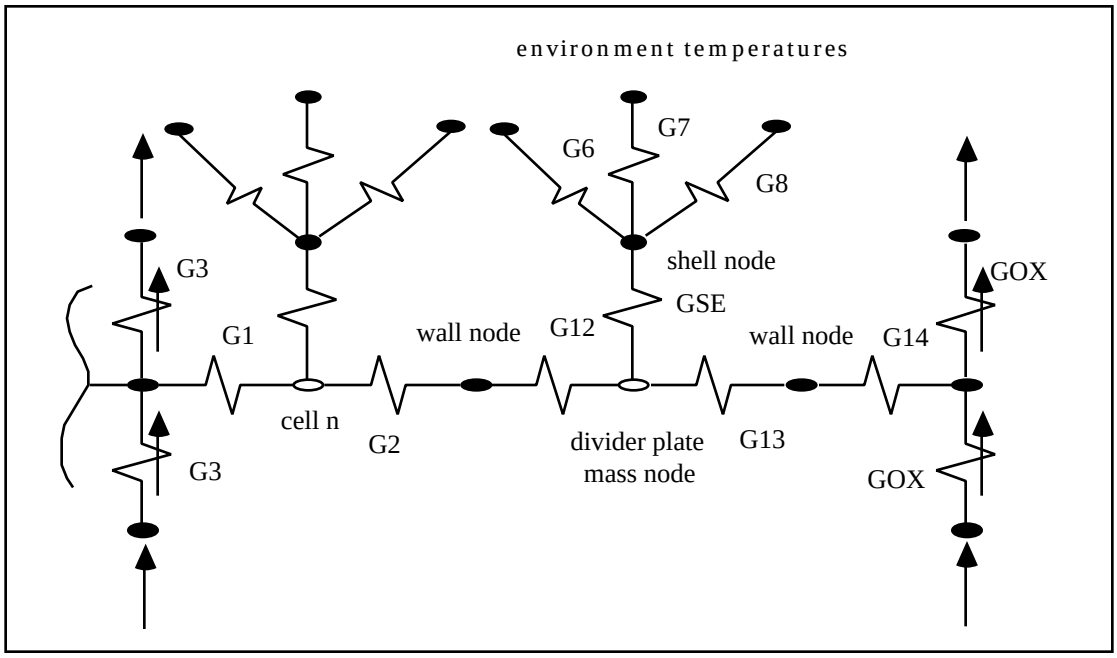


Figure 51. Divider plate node next to O_2/H_2O separator.

Mass node equations, for the divider plates and cells, were developed using the Crank-Nicolson implicit forward-backward differencing method. A block iteration was done on the system of equations. Time constants are calculated for divider plates and cells to establish the size of the minitime step for the thermal solution. Since all cells in the stack have the same capacitance and adjacent conductance values, only three time constants are determined. The routine iterates on each mass node at each minitime step until the convergence criteria for each mass node is met.

The method used for determining the heat produced by the reaction required the first law of thermodynamics for a reacting system. From reference 1, the first law for a reacting system is given by:

$$\dot{Q} + \sum_R n_i\left(\bar{h}_f^o + \Delta\bar{h}\right)_i = \dot{W} + \sum_P n_e\left(\bar{h}_f^o + \Delta\bar{h}\right)_e ,$$

where,

n = molar flow rate $\left(\frac{\text{lb-mol}}{\text{h}}\right)$

$\bar{h}_f^o$ = standard state enthalpy of formation $\left(\frac{\text{Btu}}{\text{lb-mol}}\right)$

$\Delta\bar{h}$ = change in enthalpy of formation with temperature $\left(\frac{\text{Btu}}{\text{lb-mol}}\right)$

$\dot{W}$ = work input to the system $\left(\frac{\text{Btu}}{\text{h}}\right)$

$\dot{Q}$ = resultant heat load on the module $\left(\frac{\text{Btu}}{\text{h}}\right)$.

The subscripts R and P refer to the reactants and products, respectively. The multiple of the molar flow rate and enthalpy of formation is carried out over the summation of the reactants and products. The i and e subscripts refer to the inlets and exits of the system under analysis. Applying the equation above to an electrolysis cell, the separate terms in the first law equation are evaluated as follows:

$$\sum_R n_i\left(\bar{h}_f^o + \Delta\bar{h}\right)_i$$

$$= n_{H_2O}\left(-122971.0\left(\frac{\text{Btu}}{\text{lb-mol}}\right) + \Delta\bar{h}\right),$$

$$\sum_P n_e\left(\bar{h}_f^o + \Delta\bar{h}\right)_e$$

$$= n_{H_2}\left(0.0 + \Delta\bar{h}\right) + n_{O_2}\left(0.0 + \Delta\bar{h}\right),$$

$$\dot{W} = V*I*3.41214\left(\frac{\text{Btu}}{\text{h-Watt}}\right).$$

V and I are the voltage and current of the cell, respectively. The heat load on the cell due to the reaction is then given by:

$$\dot{Q} = V*I*3.41214\left(\frac{\text{Btu}}{\text{h-Watt}}\right)$$

$$- n_{H_2}\left(0.0 + \Delta\bar{h}\right) - n_{O_2}\left(0.0 + \Delta\bar{h}\right)$$

$$+ n_{H_2O}\left(-122971.0\left(\frac{\text{Btu}}{\text{lb-mol}}\right) + \Delta\bar{h}\right).$$

This is the heat produced by the exothermic reaction at the cell core. The signs were changed in the above equation because the first law analysis of the reactants and products will show a negative heat load for an exothermic reaction. This heat load is the heat input to the cell mass node, therefore, the sign must be changed.

The recirculating loop liquid/air HX is modeled as a crossflow HX using the effectivity/NTU method. A mixed air stream and an unmixed water stream were assumed for the model. The HX effectivity is determined by:

$$e = 1.0 - e^{-\left(1.0 - e^{-\text{NTU/CRAT}}\right)/\text{CRAT}}.$$

CRAT is the ratio of $C_{\min}$ to $C_{\max}$. $C_{\min}$ is the capacity rate of the air stream and is defined by:

$$C_{\min} = \dot{m}_{\text{air}} c_{p_{\text{air}}}.$$

$C_{\max}$ is the capacity rate of the water in the recirculating loop and is defined by:

$$C_{\max} = \dot{m}_{H_2O} c_{p_{H_2O}}.$$

Number of transfer units (NTU) is a dimensionless parameter and is defined as:

$$\text{NTU} = \frac{UA}{C_{\min}},$$

where UA is the HX overall conductivity multiplied by the effective area. The value of UA must be input by the user. The exit air temperature is then calculated by:

$$T_{ao} = e(T_{in} - T_{ai}) + T_{ai},$$

where,

T_{in} = HX water inlet temperature (°F)

T_{ai} = HX air inlet temperature (°F) .

The HX outlet temperature is determined by:

$$T_o = \text{CRAT}(T_{ai} - T_{ao}) + T_{\text{in}}.$$

This is the temperature of the water reentering the cell stack.

10.2.4 Pressure Calculations

The makeup water stream, AHX inlet stream, and N_2 purge gas stream are designated as "active" within the routine. The inlet pressures for these streams must be specified by the upstream component. Outlet O_2 and H_2 stream pressures are specified in the routine by passing a value of 0.0 lb/in^2 absolute pressure for streams 2 and 3 to the PSPEC routine. This allows the pressures of these streams to be matched to the downstream components. The AHX outlet stream (stream 5) pressure must be specified in the inputs. An input value of 0.0 for the AHX outlet stream will allow this stream pressure to be set by the downstream component.

10.2.5 Input Parameters

The user input parameters for the AFSPE are shown in tables 20 and 21.

Table 20. AFSPE main edit screen.

```
CASE NAME: DEFAULT                                  AFSPE EDIT SCREEN                                  LAST UPDATED: 890428
SUBSYSTEM:              COMPONENT: DEFAULT                                                            RECORD NUMBER: 000001
 ___________________________________________________________________________________________________________________________
| ** INPUT DATA                                                                                                                |
| 10 OPERATING FLAG (0,1,2)          1.0 | 21 MAX STACK OUTLET TEMP (F)   120.00 |      ****** THERMAL NETWORK INPUTS *******      |
| 11 PRODUCTION RATE  (LB O2/HR)  0.4630 | 22 HX OPERATING TEMP BAND (F)    5.00 | 23 MASS-SHELL NODE CNDCTNCE (BTU/HR-F)     0.50 |
| 12 NO. OF ACTIVE CELLS            12.0 |                                       | 24 MASS-WALL CNDCTNCE (BTU/HR-F)           5.00 |
| 13 FARADIAC EFFICIENCY           1.000 | ** O2 LOOP H2O FEED RATES (LB/HR) **  | 25 CELL COOLING EFFECTIVITY (0-1)          0.90 |
| 14 INDIVIDUAL CELL VLTGE (V)     1.600 |                NORMAL     EMERGENCY   | 28 SHELL-ENV CONDUCTION (BTU/HR-F)        0.500 |
| 15 PROTONICLY PUMPED H2O RATIO    8.00 |    HIGH     26   6.00     34   7.50   | 29 SHELL-ENV CONVECTION (BTU/HR-F)        0.050 |
| 16 INITIAL CELL TMPRATURES (F)   75.00 |    LOW      27   4.50     35   6.00   | 30 FORM FACTOR*AREA*EMISSIVITY (FT^2)   0.05000 |
| 17 O2 LOOP/H2O FLOW (LB/HR)     240.00 |                                       | 31 SINGLE CELL CAPACITANCE (BTU/F)      0.50000 |
| 18 HX FAN FLOW RATE (LB/HR)     200.00 |                                       | 32 MAX # OF LOOPS                         200.0 |
| 19 AIR/LIQ HX UA (BTU/HR-F)      50.00 |                                       | 33 RELAXATION                          0.000100 |
| 20 HX OUT STREAM PRESS (PSI)     0.000 | 7 POWER  (WATTS)    0.00              |                                                 |
|    (0.0 IF DOWNSTREAM COMPONENENT      | 8 WEIGHT  (LBM)     0.00              |                                                 |
|           SETS THE PRESSURE)           | 9 VOLUME  (FT^3)    0.00              | ** OUTPUT/BENCHMARK DATA (^L TO OPEN) _         |
|_______________________________________|_______________________________________|_________________________________________|
| ** ENTHALPY OF FORMATION CURVE FIT DATA                                                                                      |
|      DEPNDNT VARIABLE                   INDEPNDNT VARIABLE  |       ^3       |      ^2        |      ^1        |       ^0       |
|  O2 DELTA ENTHLPY (BTU/LB-MOL)          TEMPERATURE (degR)  |  0.344430E-06  |  -.224671E-03  |   6.98300      |  -3735.00      |
|  H2 DELTA ENTHLPY (BTU/LB-MOL)          TEMPERATURE (degR)  |  0.188000E-05  |  -.246000E-02  |   7.55000      |  -3625.00      |
|  H2O VAPOR DELTA ENTHLPY (BTU/LB-MOL)   TEMPERATURE (degR)  |  0.180000E-06  |  0.176000E-04  |   7.88000      |  -4262.00      |
|_____________________________________________________________________________________________________________________________|
```

Table 21. AFSPE output edit screen.

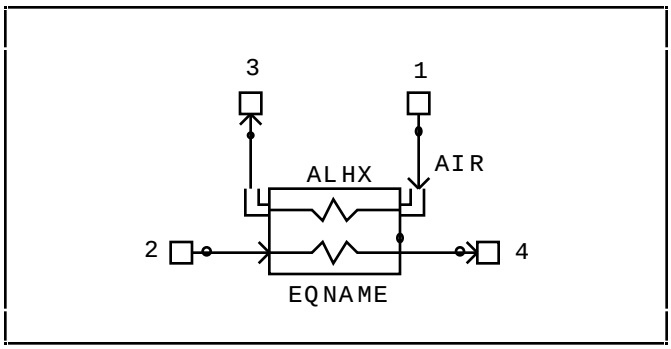
```
CASE NAME: DEFAULT                                  AFSPE OUTPUT SCREEN                                  LAST UPDATED: 890428
SUBSYSTEM:              COMPONENT:                                                                      RECORD NUMBER: 000001
 ___________________________________________________________________________________________________________________________
| ** OUTPUT DATA                                                                  | ** BENCHMARK DATA                                |
| AVERAGE CELL TEMPERATURE (F)     0.00   O2 STREAM-O2 MASS FRAC  0.0000 |                                  MIN       NOM       MAX    |
| AVERAGE SHELL TEMPERATURE (F)    0.00   O2 STREAM-H2O MASS FRAC 0.0000 | H20 RATE IN     (LBM/HR)    0.000     0.000     0.000  |
| STACK EXIT FLOW TEMP (F)         0.00   H2 STREAM-H2 MASS FRAC  0.0000 | H2/H2O RATE OUT (LBM/HR)    0.000     0.000     0.000  |
| O2 OUTLET FLOW  (LB/HR)        0.0000   H2 STREAM-H2O MASS FRAC 0.0000 | O2/H2O RATE OUT (LBM/HR)    0.000     0.000     0.000  |
| H2 OUTLET FLOW  (LB/HR)        0.0000                                   |                                                  |
|_________________________________________________________________________|_________________________________________________|
```

Reference

1. Van Wylen and Sonntag: “Fundamentals of Classical Thermodynamics.” Second edition, John Wiley and Sons Inc., 1973, pp. 690–697.

10.3 ALHX—Air/Liquid Heat Exchanger Simulation Description

This routine simulates the thermal control function of an air/liquid HX (ALHX). HX analysis is based on the effectiveness-NTU method. HX effectiveness is calculated by the routine utilizing a user-specified overall heat conductance (UA) with cross-flow (both fluids unmixed), counterflow, or parallel-flow equations depending on the desired application. The ALHX does not exchange heat with its environment. The ALHX does not generate flow; therefore, an active component must be upstream of each inlet. Pressure drop through the ALHX is determined by user-supplied equivalent line lengths and equivalent line diameters. The ALHX data base parameters are shown in table 22.



Discussion

The ALHX routine is based on the effectiveness-NTU analysis approach of HX performance. This method uses specified inlet flow rates, specific heats, and temperatures, in conjunction with a user-specified overall heat conductance (*UA*). It is assumed that there is negligible heat transfer between the exchanger and its surroundings, as well as negligible potential and kinetic energy changes, which allow use of the appropriate energy balance equations. The resultant outlet temperatures and heat transfer characteristics are determined as functions of the above data and an internally calculated HX effectiveness. Three distinct types of HX simulation are currently available; parallel-flow, counterflow, and cross-flow (both fluids unmixed). The HX effectiveness relationships for these geometrically dissimilar flow arrangements are given as follows:

$$\varepsilon = \frac{1-\exp[-\mathrm{NTU}(1+C_r)]}{1+C_r}\,, \qquad \text{(parallel-flow)}$$

$$\varepsilon = \frac{1-\exp[-\mathrm{NTU}(1-C_r)]}{1-C_r\exp[-\mathrm{NTU}(1-C_r)]}\,, \qquad \text{(counterflow)}$$

$$\varepsilon \approx 1-\exp\{\frac{1}{C_r}\mathrm{NTU}^{0.22}[\exp(-C_r\mathrm{NTU}^{0.78})-1]\}\,, \qquad \text{(cross-flow)}$$

where,

$\mathrm{NTU} = UA/C_{\min}$

$C_r = C_{\min}/C_{\max}$

$C_{\min} = \dot{m}C_p$, minimum fluid heat capacity rate, Btu/h/°F

$C_{\max} = \dot{m}C_p$, maximum fluid heat capacity rate, Btu/h/°F

UA = overall heat conductance, Btu/h/°F.

A special case arises for the counterflow HX when the inlet heat capacity rates are equal. In this situation the effectiveness equation reduces to the following:

$$\varepsilon = \frac{\mathrm{NTU}}{1+\mathrm{NTU}}\,.$$

Now the effectiveness and the maximum possible heat transfer rate can be related with the following equations to calculate the actual heat transfer within the exchanger:

$$q_{\max} = C_{\min}(T_{h,i}-T_{c,i})\,,$$

$$q = \varepsilon\, q_{\max}\,,$$

where,

$q_{\max}$ = maximum possible heat transfer rate, Btu/h

q = actual heat transfer rate, Btu/h

$T_{h,i}$ = hot fluid inlet temperature, °F

$T_{c,i}$ = cold fluid inlet temperature, °F.

It is now a simple matter to determine the outlet temperatures from the overall energy balances.

It should be noted that the above effectiveness equation for the cross-flow HX type can be used with confidence only when $C_r \approx 1.0$. Therefore, the routine will warn the user with a message to the screen if C_r is greater than 1.15 or less than 0.85 for the cross-flow geometry.

The above methods of solution may be generalized to include other flow geometries via proper modification of the effectiveness equation. The derivations of the above effectiveness relationships are provided in references 1 and 2.

Pressure drop through the ALHX is found by conventional pressure drop calculation methods utilizing user-supplied equivalent lengths and equivalent diameters. See section 8.2 for a detailed explanation of the pressure drop calculations.

References

1. Incropera, F.P., and DeWitt, D.P.: "Fundamentals of Heat Transfer." Wiley, 1981.

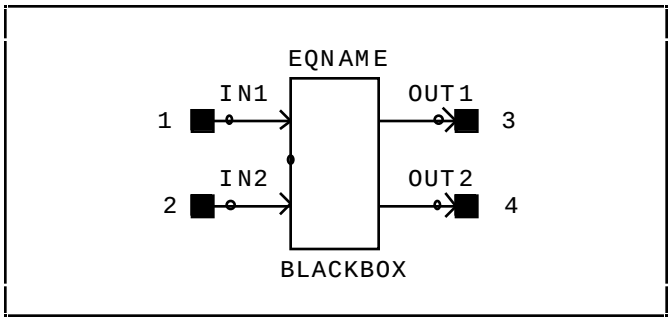
2. Kays, W.M., and London, A.L.: "Compact Heat Exchangers." McGraw-Hill, 1984.

Table 22. ALHX edit screen.

```
CASE NAME: DEFAULT              ALHX EDIT SCREEN              LAST UPDATED: 890215
SUBSYSTEM:                     COMPONENT:                    RECORD NUMBER: 000001
_____________________________________________________________________________
|            ** GENERAL INPUT DATA **                |   ** RESOURCE TRACKING **  |
| 10 OVERALL UA                (BTU/HR/F)  350.00    |                            |
| 15 FLOW: 1=CNTR,2=PARAL,3=CROSS            1.00    | 7 POWER  (WATTS)    0.00   |
|                                                    | 8 WEIGHT   (LBM)    0.00   |
|         ** PRESSURE DROP PARAMETERS **             | 9 VOLUME  (FT^3)   0.000   |
| 11 AIR LOOP EQUIVALENT LENGTH     (FT)     1.00    |                            |
| 12 AIR LOOP EQUIVALENT DIAMETER   (IN)    6.000    |                            |
| 13 LIQ LOOP EQUIVALENT LENGTH     (FT)     1.00    |                            |
| 14 LIQ LOOP EQUIVALENT DIAMETER   (IN)    1.000    |                            |
|                                                    |                            |
|                                                    |                            |
|                                                    |                            |
|____________________________________________________|____________________________|
|                 ** BENCHMARK DATA **               |     ** OUTPUT DATA **      |
|                      MIN        NOM        MAX    |                            |
| HEAT TRANS (W)   0.000E+00  0.000E+00  0.000E+00| 17 HEAT TRANS  (W)    0.00 |
|____________________________________________________|____________________________|
```

10.4 BLACKBOX—BLACKBOX Simulation Description

This component was developed to allow the user to create specialized components not currently modeled. The number of inlet and outlet streams are entered by the user via the LO command and the graphical icon is adjusted accordingly. A maximum of eight streams is allowed with four inlet and four outlet streams. The user is responsible for providing the FORTRAN logic required to simulate the process under consideration. A template for developing this code is provided in a file named BBOX.FOR, which is located in the code directory. The BBOX.FOR routine should be copied into the user's directory and renamed to reflect the specialized component/process. The modified routine may be appended to the CASENAME.FOR file for convenience. Up to 25 BLACKBOX components may be used in a single model. See section 7.6.3 for further information.



The location format of the BLACKBOX device is as follows:

LO;BLACKBOX;EQNAME;TYPE:NINLETS;NOUTLETS

where TYPE is between 1 and 25, NINLETS and NOUTLETS are the number of inlet and outlet fluid streams, respectively. The stream labels (up to four characters long) will be solicited from the user after the LO command has been entered.

10.4.1 General Discussion

Generally, each component routine has four phases which must be addressed. These phases are designated as:

A. The component data load phase (100 block)

B. Begin time step/iteration phase (200 block)

C. Wrap-up time step (300 block)

D. Post simulation component wrap-up (400 block).

The first phase is used to load component data from the component data files. A generic component data file was created with blank records to allow the user to insert data as required. The second phase starts the time steps and will continue in this phase until the system convergence is obtained or the maximum number of loops is exceeded. The third phase is used for post time step adjustments required for components such as tanks and cabins. This phase is used to adjust accumulated mass and set old temperatures to the new values. The final phase is used to write the results back to the component data files for bench marking and summarizing consumables.

The common blocks in the "COMPCOM.INC" file are required and should be included in the BLACKBOX (OPS7) subroutine. If the component that is being developed is an active component, i.e., generates flow similar to a fan or pump, then the WFLOW parameter must be set to the desired flow rate; otherwise, set the value to 0.0. The program will automatically set all of the required internal points and parameters and will establish the arrays that are used to check for system convergence.

Incoming constituents, and their properties, are obtained through the C and PRO arrays, respectively. The program establishes pointers to these arrays at the beginning of the solution. The incoming constituents can be found from the "NPT(X)" array where X is the relative stream number as defined by the user in the LOcate command. That is, input stream 1 is NPT(1) and outlet stream 8 is NPT(8). The PRO array elements are defined in section 7.2 and the format is:

PRO(PROPERTY, Relative number, Stream number).

PROPERTY is an integer number from 1 to 8 that represents the appropriate property (i.e., 1 for pressure, 2 for temperature, etc., as shown in section 7.2). The relative number is passed through a common block in the variable IEQ. The stream number is the component stream number. An example of both the PRO and C array references is shown below.

TEMP = PRO (2,IEQ,1)
MDOT_WATER = C(1, NPT(1)) * C(8,NPT(1))

TEMP is the temperature of stream 1 and MDOT_WATER is the mass flow rate of water for stream 1.

Heat loads or environmental temperatures can be passed through CTEMP(NCAB,X), where NCAB is the appropriate cabin determined by the program from the assignment process (ASSIGN), and X is 1 for cabin air, 2 for cabin structure, and 3 for cabin mean radiative environment. Cabin air, structure, and/or mean radiant environments are available through the CTEMP(NCAB), CSTR(NCAB) and/or CRAD(NCAB) arrays, respectively.

In accordance with the mass conservation equations, the constituent values in the “C” arrays must be passed, processed, or accumulated by the component. The first location in the “C” array is reserved for the total mass flow rate of the stream under consideration.

System convergence is checked by the program through utility routines (KHECK), which the user must perform if required (recommended). These routines should be the final action prior to a return in the 200 block of code.

10.4.2 Input Parameters

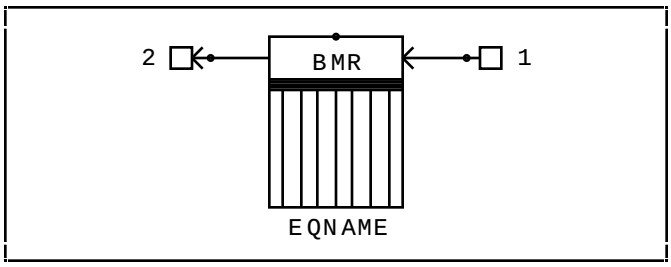
A component data file is provided for the BLACKBOX, but the definition and use of the input parameters are not established; therefore, all performance parameters must be defined by the user. A generic screen is provided with 30 input parameters locations available as shown in table 23. Six output and three benchmark outputs are provided for as well. If the user requires additional parameters, the OPS7 logic block should be used to store or access the data. See section 7.1 for a more detailed discussion of the operations blocks.

Table 23. BLACKBOX edit screen.

```
CASE NAME: BLACKBOX                                  BLACKBOX EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM: DEFAULT       COMPONENT: VALUES                                                                RECORD NUMBER: 000001
_________________________________________________________________________________________________________________________________
|                                                   ** GENERAL INPUT DATA **                                                    |
| 10 INPUT CONSTANT  #1            0.000E+00 | 20 INPUT CONSTANT #11          0.000E+00 | 30 INPUT CONSTANT #21          0.000E+00 |
| 11 INPUT CONSTANT  #2            0.000E+00 | 21 INPUT CONSTANT #12          0.000E+00 | 31 INPUT CONSTANT #22          0.000E+00 |
| 12 INPUT CONSTANT  #3            0.000E+00 | 22 INPUT CONSTANT #13          0.000E+00 | 32 INPUT CONSTANT #23          0.000E+00 |
| 13 INPUT CONSTANT  #4            0.000E+00 | 23 INPUT CONSTANT #14          0.000E+00 | 33 INPUT CONSTANT #24          0.000E+00 |
| 14 INPUT CONSTANT  #5            0.000E+00 | 24 INPUT CONSTANT #15          0.000E+00 | 34 INPUT CONSTANT #25          0.000E+00 |
| 15 INPUT CONSTANT  #6            0.000E+00 | 25 INPUT CONSTANT #16          0.000E+00 | 35 INPUT CONSTANT #26          0.000E+00 |
| 16 INPUT CONSTANT  #7            0.000E+00 | 26 INPUT CONSTANT #17          0.000E+00 | 36 INPUT CONSTANT #27          0.000E+00 |
| 17 INPUT CONSTANT  #8            0.000E+00 | 27 INPUT CONSTANT #18          0.000E+00 | 37 INPUT CONSTANT #28          0.000E+00 |
| 18 INPUT CONSTANT  #9            0.000E+00 | 28 INPUT CONSTANT #19          0.000E+00 | 38 INPUT CONSTANT #29          0.000E+00 |
| 19 INPUT CONSTANT #10            0.000E+00 | 29 INPUT CONSTANT #20          0.000E+00 | 39 INPUT CONSTANT #30          0.000E+00 |
|                                            |                                          |                                          |
| 7 POWER (WATTS)                  0.000E+00 | 8 WEIGHT (LBM)                 0.000E+00 | 9 VOLUME (FT^3)                0.000E+00 |
|____________________________________________|__________________________________________|__________________________________________|
|                      ** BENCHMARK DATA **                  |                          ** OUTPUT DATA **                                |
|                                                            |                                                                           |
| BENCHMARK #1   0.0000E+00  #2  0.0000E+00  #3  0.0000E+00 | 40 OUTPUT CONSTANT #1   0.000E+00  43 OUTPUT CONSTANT #4  0.000E+00  |
| BENCHMARK #4   0.0000E+00  #5  0.0000E+00  #6  0.0000E+00 | 41 OUTPUT CONSTANT #2   0.000E+00  44 OUTPUT CONSTANT #5  0.000E+00  |
| BENCHMARK #7   0.0000E+00  #8  0.0000E+00  #9  0.0000E+00 | 42 OUTPUT CONSTANT #3   0.000E+00  45 OUTPUT CONSTANT #6  0.000E+00  |
|____________________________________________________________|___________________________________________________________________________|
```

10.5 BMR—Body Mounted Radiator Simulation Description

This routine simulates the thermal control behavior of a BMR. The BMR routine has a built-in bypass function that allows the user to specify a desired downstream mixture temperature. This mixture temperature is achieved by manipulating the amount of fluid allowed to bypass the radiator. This routine has a built-in error trap that is activated if the radiator area is not large enough to achieve the desired mixture temperature. The BMR does not generate flow; therefore, an active component must be upstream of the inlet. Also, if the fluid inlet temperature is less than the specified downstream mixture temperature, the radiator is bypassed entirely. The pressure drop through the radiator is found utilizing user-supplied equivalent line lengths and equivalent line diameters for both flow through the radiator and flow through the bypass. The BMR data base parameters are shown in table 24.



Discussion

The BMR routine simulates a two-phase radiator that interfaces with a contact HX. A schematic of the composite thermal network, including bypass connections, is shown in figure 52. The contact HX is similar to the CNHX component and is simulated by a delta network. The radiator segment is assumed to be designed for the specific application (i.e., required heat transfer rate and operating temperature). Phenomena normally associated with two-phase heat pipe radiators, such as variations in fluid properties, capillary forces, boiling, and sonic limitations are ignored. The user can modify the performance of the radiator by changing either the effective sink temperature or the radiative conductance. These operations may be implemented into supplemental OPS logic and dynamically computed as a function of time, properties, etc., if desired. An example of a similar procedure is provided in figure 53. Radiator outlet temperature is a user-specified quantity and is achieved via proper manipulation of the bypass flow rate. Determination of the bypass setting occurs iteratively with the nodal temperature relaxation procedure. If an unobtainable outlet temperature is specified (based on physical constants, conditions, and boundaries), the nodal network will be solved with the bypass setting locked full bypass or no bypass, depending on the situation. Appropriate error messages will be printed to the screen and the resultant nodal temperatures are passed to the system logic.

Pressure drop through the BMR is found by conventional pressure drop calculation methods utilizing a user-supplied equivalent length and equivalent diameter. See section 8.2 for a detailed explanation of the pressure drop calculations.

Table 24. BMR edit screen.

```
CASE NAME: DEFAULT              BMR EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                    COMPONENT:                  RECORD NUMBER: 000001
_____________________________________________________________________________
|          ** GENERAL INPUT DATA **          |                                |
| 18 OUTLET SET POINT TEMP   (F)     0.00    |     ** RESOURCE TRACKING **    |
|                                            | 7 POWER    (WATTS)          0.00 |
|     ** THERMAL CONDUCTANCE DATA **         | 8 WEIGHT   (LBM)            0.00 |
| 10 THROUGHPUT COND  (BTU/HR/F)    0.900    | 9 VOLUME   (FT^3)          0.000 |
| 11 FLUID-WALL COND  (BTU/HR/F)    1.000    |                                |
| 12 RADIATIVE F*A*E      (FT^2)    0.900    |   ** PRESSURE DROP PARAMETERS **|
|                                            | 16 EQUIVALENT LENGTH (FT)   10.00 |
|      ** THERMAL INTERFACE DATA **          | 17 EQUIVALENT DIA    (IN)   0.750 |
| 13 ENVIRONMENTAL SINK TEMP (F)  -100.00    |                                |
|                                            |                                |
|  ** SUB-SOLUTION CONTROL PARAMETERS **     |                                |
| 14 MAX LOOP COUNT                 1355.0   |                                |
| 15 TEMPERATURE RELAXATION  (F)     0.80    |                                |
|____________________________________________|________________________________|
|                ** BENCHMARK DATA **                |     ** OUTPUT DATA **  |
|                      MIN        NOM        MAX     |                        |
| HEAT TRANSFER (W)  0.00E+00   0.00E+00   0.00E+00|19 HEAT REJECT (W)     0.0 |
| OUTLET TEMP   (F)  0.00E+00   0.00E+00   0.00E+00|                        |
|_____________________________________________________|________________________|
```

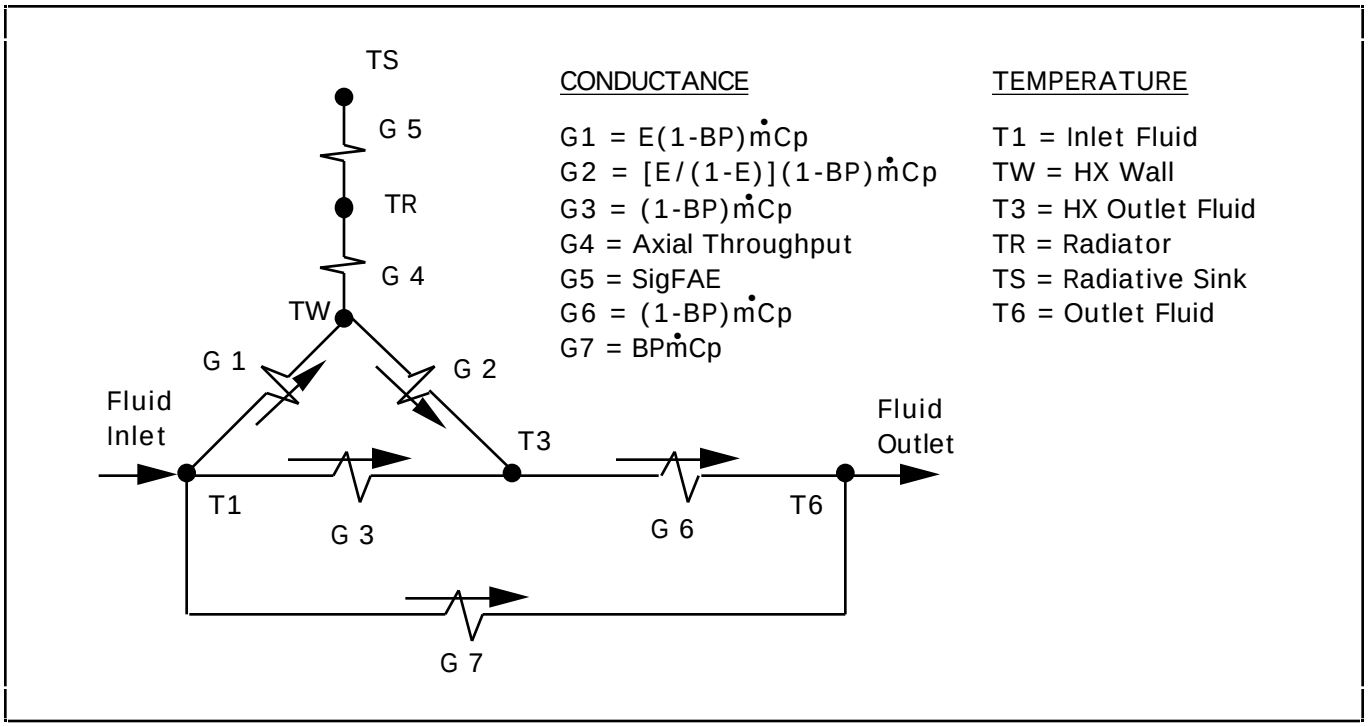


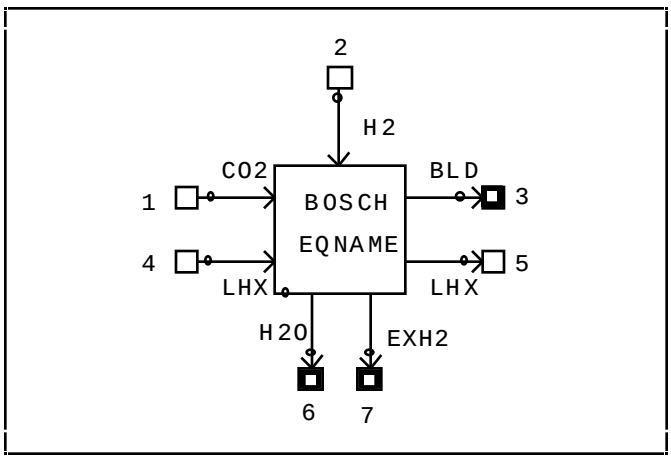
Figure 52. BMR thermal network.

```
C
C     OPS3 BLOCK-CALLED EVERY ITERATION
C
  DATA FINEFF/.90,.95,.90,.85,.80,.70/
  DATA INTEMP/45,50,55,60,65,70/
C
C     CONTROL LOGIC FOR VARYING RADIATOR
C     FIN EFFICIENCY AS A FUNCTION OF
C     FLUID INLET TEMPERATURE
C
C     GET INLET TEMPERATURE
  CALL GETT('BMRX',1,TBMRIN)
C     CALL STANDARD INTERPOLATION ROUTINE
  CALL INTER2(TBMRIN,INTEMP,FINEFF,BMREFF)
C     SET FIN EFFICIENCY
  CALL SETK('BMRX',14,BMREFF)
C
C     OPS4 BLOCK-CALLED AT END OF TIME STEP
C
  DATA SINKT/-100,-105,-110,-115,-112,-107/
  DATA ORB/0,15,30,45,60,75 /
C
  IF(ORBTIME .GT. 90) ORBTIME = 0
  ORBTIME = ORBTIME + STEP
  CALL INTER2(ORBTIME,ORB,SINKT,TEMP)
  CALL SETK('BMRX',16,TEMP)
```

Figure 53. Sample logic for modifying BMR fin efficiency and sink temperature.

10.6 BOSCH—BOSCH Simulation Description

This component is used to simulate heat and mass transfer characteristics of the BOSCH carbon dioxide reduction process, which includes a catalytic reactor, regenerative HX, condensing HX, and compressor. A single reactor bed is simulated and an instantaneous cartridge changeout is performed. The component calculates the amount of H_2 required to maintain stoichiometric compositions. If there is excess H_2 in the makeup stream, it is directed through stream 7. Insufficient H_2 reduces the CO_2 conversion rate. Accumulated carbon (per cartridge) is calculated and used to determine process recycle flow rates. Active cooling is required for condenser simulation and must be supplied by the user.



10.6.1 General Discussion

The BOSCH chemical reaction is exothermic and occurs in the presence of an iron catalyst. A schematic of the modeled process is shown in figure 54. The overall stoichiometric equation for this reaction is given as follows:

$$CO_2 + 2H_2 \xrightarrow[\text{cat}]{} C(s) + 2H_2O \ .$$

In practice, however, single pass efficiencies through the Bosch reactor tend to be less than 10 percent. Thus, to provide continuous reduction of carbon dioxide into solid carbon and water, the process gas mixture is recycled. The reduction occurs through the following intermediate reactions:

A. $CO_2 + H_2 \leftrightarrow CO + H_2O$

B. $CO + H_2 \leftrightarrow C + H_2O$

C. $CO + CO \leftrightarrow C + CO_2$

D. $C + 2H_2 \leftrightarrow CH_4$.

The following assumptions were made during the development of the BOSCH routine.

A. Startup operation is not modeled due to a lack of low temperature reaction kinetics test data for verification. At simulation initiation, a “new” cartridge is in the reactor, and a mixture of CO_2-H_2 is present in the system.

B. Perfect temperature control of the reactor mass is assumed and the necessary heater requirements are calculated.

C. The catalyst temperature is set equal to the reactor mass temperature.

D. The four key reactions are modeled as separate bi-directional reactions competing for the same reactants.

E. The reactions are mass transfer controlled with chemical equilibrium existing at the catalyst surface.

F. Due to the complex chemical and physical phenomena occurring at the catalyst surface, the forward and equilibrium reaction rate constants are determined empirically through correlation to existing test data.

10.6.2 BOSCH Carbon Dioxide Reduction Process

The BOSCH CO_2 reduction process consists of a catalytic reactor, regenerative HX, condensing HX, and a compressor. A schematic of the modeled process is shown in figure 56.

The process rate (speed at which carbon dioxide is reduced to solid carbon and water) is primarily controlled by recycle flow rate, catalyst surface area, operating temperature, pressure, and carbon loading. For a given set of system temperatures, pressures, flow rates, and catalyst configuration, the reduction process will reach

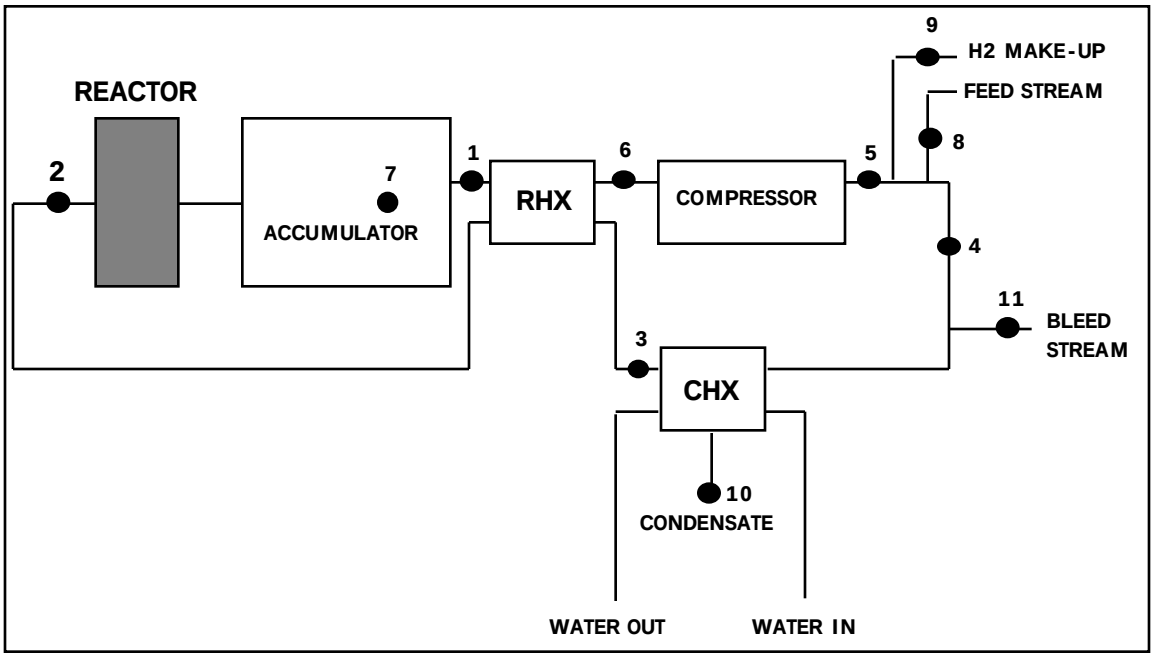


Figure 54. BOSCH thermal network.

equilibrium operating conditions. Thus, the accumulation of carbon in the catalyst surface is modeled as the controlling factor governing performance degradation of the BOSCH component routine.

The mass and thermal conductances through the ash to the catalyst surface are set arbitrarily large when there is no solid carbon present, i.e., at the beginning of a simulation and/or cartridge changeout. Once carbon accumulation begins, it is assumed to form cylindrical shells around individual catalyst "strands." The outside radius of this carbon shell, r_o, is given by:

$$r_o = \sqrt{r_i^2 + \frac{M_c}{\pi \rho \alpha \ell_c}} ,$$

where,

r_i = radius of catalyst strand, ft

ℓ_c = effect length of catalyst strand, ft

M_c = mass of accumulated carbon, lbm

ρ = carbon density, lbm/ft^3

α = carbon porosity factor.

Based on this effective carbon radius, r_o, the mass and thermal conductances through the ash to the catalyst, G_m and G_t, respectively, are given by:

$$G_m = \frac{2\pi \ell_c K_d (1-\alpha)}{\ln\left(\frac{r_o}{r_i}\right)} ,$$

and

$$G_t = \frac{2\pi \ell_c K_t (1-\alpha)}{\ln\left(\frac{r_o}{r_i}\right)} ,$$

where,

K_d = reactor gas diffusion coefficient, ft^2/h

K_t = reactor gas effective thermal conductivity, $\frac{\text{Btu}}{\text{h}\bullet\text{ft}^2\bullet\text{F}}$.

The vapor densities of each of the constituents at the ash surface, $\rho_A(i)$, are computed as follows:

$$\rho_A(i) = \frac{\varepsilon_m \dot{V} \rho_I(i) + Q_R(i)}{\varepsilon_m \dot{V}} ,$$

where,

ε_m = reactor mass transfer effectivity

$\dot{V}$ = volumetric flow rate through reactor, ft^3/h

$\rho_I(i)$ = inlet vapor density (mass/reactor volume), lbm/ft^3

$Q_R(i)$ = constituent generation rate (calculated iteratively), mol/h,

and the subscript i defines the constituents as follows:

1 – carbon dioxide (CO_2)
2 – hydrogen (H_2)
3 – methane (CH_4)
4 – carbon monoxide (CO)
5 – water (H_2O)
6 – nitrogen (N_2).

The catalyst surface vapor densities, $\rho_c(i)$, are calculated from the vapor densities at the ash surface, the species generation rates, and the mass conductivity.

$$\rho_c(i) = \rho_a(i) + \frac{Q_r(i)}{G_m} ,$$

The outlet vapor densities, $\rho_o(j)$, are computed from the convective delta network equations as follows:

$$\rho_o(j) = \frac{\beta\ \rho_A(i) + \dot{V} \rho_I(j)}{\beta + \dot{V}} ,$$

where,

$$\beta = \frac{\varepsilon_m \dot{V}}{1 - \varepsilon_m} .$$

Upon converting vapor densities to partial pressures, the conversion rates of the intermediate reactions, Rj, are calculated as follows:

$$R_1 = KF_1 A_{cat} \rho_c(1) \rho_c(2) - KR_1 A_{cat} \rho_c(4) \rho_c(5) ,$$

$$R_2 = KF_2 A_{cat} \rho_c(4) \rho_c(2) - KR_2 A_{car} \rho_c(5) ,$$

$$R_3 = KF_3 A_{cat} [\rho_c(4)]^2 - KR_3 A_{car} \rho_c(5) ,$$

$$R_4 = KF_4 A_{car} [\rho_c(2)]^2 - KR_4 A_{cat} \rho_c(3) ,$$

where,

KFj = forward reaction rates

KRj = reverse reaction rates

A_{cat} = catalyst effective surface area

A_{car} = solid carbon effective surface area.

Finally, species molar generation and consumption rates, Q_R, are given by:

$$Q_R\ (CO_2) = -R_1 + R_3 ,$$

$$Q_R\ (H_2) = -R_1 - R_2 - 2\ R_4 ,$$

$$Q_R\ (CH_4) = -R_4 ,$$

$$Q_R\ (CO) = -R_1 - 2\ R_3 + R_1 ,$$

$$Q_R\ (H_2O) = -R_1 + R_2 ,$$

$$Q_R\ (C) = -R_2 + R_3 - R_4 .$$

Based on these molar rates, the overall heat of rejection, Q_t is given by:

$$Q_t = \sum_{i=1}^{6} Q_{R_i} H_{F_i} ,$$

where,

H_{F_i} = heat of formation.

10.6.3 Regenerative HX

The RHX has associated with it an effectivity factor, ε_{RHX}, defined as

$$\varepsilon_{RHX} = \frac{T_2 - T_3}{T_2 - T_6} ,$$

where,

T_2 = RHX hot side inlet (fig. 56)

T_3 = RHX hot side outlet

T_6 = RHX cold side inlet.

A simple energy balance applied to the RHX yields an expression for the reactor inlet temperature, T_1, as a function of the heat transfer rate, $\dot{Q}$, which is defined by the effectivity equation.

$$\dot{Q} = \dot{m}_2 C_{p2} (T_2 - T_3) \ ,$$

where,

$\dot{m}_2$ = recycle flow rate minus carbon generation rate

C_{p2} = constant pressure specific heat, hot side RHX.

Therefore,

$$T_1 = T_6 + \frac{\dot{Q}}{\dot{m}_6 C_{p6}} \ ,$$

where,

$\dot{m}_6$ = recycle flow rate

C_{p6} = constant pressure specific heat, cold side.

10.6.4 Condensing HX

A fundamental description of the CHX code can be found in reference 1. The BOSCH routine assumes that water vapor created during the CO_2 reduction process leaves the condenser as a liquid at the condenser primary side exit temperature, T_4.

10.6.5 Bleed Stream

Bleed stream is modeled as a continuous flow of reactants out of the Bosch control volume. The bleed stream is designed to prevent the reactor pressure from exceeding the setpoint pressure. If the reactor pressure is predicted to be below the setpoint pressure, no bleed flow is generated. The reactor pressure is then allowed to vary as a function of the net generation or consumption of gaseous moles within the control volume. If bleed stream flow is necessary, the composition of the bleed stream is equivalent to that of the CHX secondary side outlet air stream.

10.6.6 Compressor

A compressor controller adjusts the rotor speed in response to an increase in downstream pressure in accordance with the following relation based on actual controller transfer function constants (Life Systems, Inc.):

$$\text{RPM} = 1{,}086.675 + 16.553\, P \ ,$$

where,

P = reactor pressure (lb/in^2 absolute).

The volumetric flow of the compressor as a function of rotor speed was estimated from vendor data to be 0.001652 CFM/RPM. The recirculation mass flow rate is then calculated based on the current reactor pressure and composition.

The pressure drop contribution from the reactor as a function of carbon loading is calculated using the Burke-Plummer equation.[5]

$$\frac{150}{Re_p} + 1.75 = \frac{GcD\Delta P\varepsilon^3}{L\ Vbs^2 Q\rho(1-\varepsilon)} \ ,$$

where,

Re_p = Reynolds number $\dfrac{D\ V_{bs}\rho}{\mu(1-\varepsilon)}$

Gc = Gravitation constant

D = effective diameter

ΔP = pressure drop

ε = packed bed void fraction

L = flow length

V_{bs} = superficial velocity

ρ = gas density.

The effective diameter for long filaments is $3r$, where the radius, r, is determined by the

carbon load on the catalyst as described previously. Using this relation, the pressure drop contribution from the reactor bed was found to be small except very near the maximum bed loading point. The data from CMIF testing, however, suggests that a significant constant drop of 2 to 7 lb/in^2 absolute is seen in the actual hardware.

The lower pressure drops occur at lower system pressures (typical of stoichiometric feed streams) and the higher pressure drops at higher system pressures (typical of feed streams containing inerts).

Using Darcy's formula, an additional system pressure drop contribution is given by:

$$\Delta P = \frac{K\dot{m}^2}{\rho}$$

where,

ΔP = observed system pressure drop (not a function of bed loading)

K = empirically derived constant, 0.00340

$\dot{m}$ = mass flow rate

ρ = gas density.

Adding the reactor and system pressure drop contributions provides results which reasonably match the pressure dependent offset and rapid rise near cartridge capacity. The maximum cartridge loading point presented in the CMIF test data, as indicated by a sharp rise in ΔP, varies significantly. These variations may be due to differences in the actual cartridges or differences in the physical nature of the carbon deposited as a function of test conditions and/or reduction rates.

Using a simple expression, the air temperature at the compressor outlet, T_6, can be calculated as follows:

$$T_6 = T_5 + \frac{Q_c}{\dot{m}_5 C_{p5}},$$

where,

$\dot{m}_5$ = recycle flow plus process gas flow plus hydrogen makeup flow

T_5 = process gas mixture temperature

C_{p5} = mass averaged constant pressure specific heat, process gas mixture

Q_c = compressor waste energy plus compression work (Btu/h).

10.6.7 Input Parameters

The user input parameters for the Bosch reactor are shown in table 25. A majority of the input parameters are determined by given hardware configurations. The default data has been derived from published literature when possible. Any unavailable data was determined through parametric experimentation or estimated as necessary.

References

1. Computer Aided System Engineering and Analysis (CASE/A) BOSCH Component Routine Verification Analysis, MDC 5181.
2. Space Station Cluster Closed Loop Oxygen Recovery Models - Carbon Dioxide Reduction, ECLSS Integration Analysis, MDC W5062-1.
3. Performance Characterization of a Bosch CO_2 Reduction Subsystem, NASA CR-152342, Life Systems Inc.
4. Carbon Dioxide Reduction Processes for Spacecraft ECLSS: A Comprehensive Review, SAE Technical Paper Series, 881042, 18th Intersociety Conference on Environmental Systems, San Francisco, CA, July 11-13, 1988.
5. Momentum, Heat, and Mass Transfer, Bennett and Myers, McGraw-Hill, 1962.

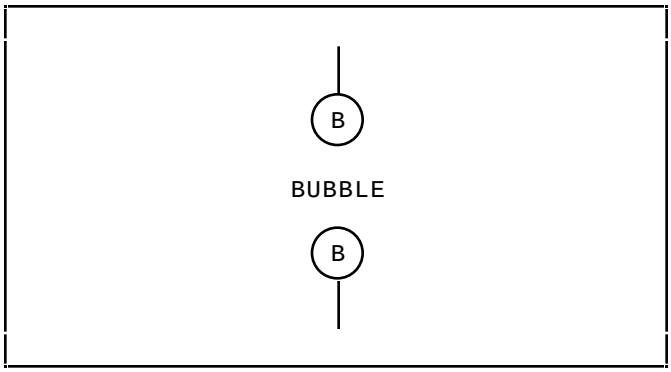
Table 25. BOSCH edit screen.

```
CASE NAME: BOSCH                                   BOSCH EDIT SCREEN                                   LAST UPDATED: 900328
SUBSYSTEM: DEFAULT       COMPONENT: VALUES                                                           RECORD NUMBER: 000001

|11 RAD CATALYST STRAND  (FT) 0.200E-03 |19 REACT HEAT TRANS EFF   0.95000 | REACTION RATE CONSTANTS       FORWARD       EQUILIBRIUM |
|13 CANISTER RADIUS        (IN)  3.5000 |20 REACT MASS TRANS EFF   0.95000 | CO2 +  H2 <-> CO + H2O  29 .22500E-02   33 .56200      |
|14 CANISTER LENGTH        (IN) 12.0000 |21 REGENERATIVE HX EFF    0.85000 | CO  +  H2 <-> C  + H2O  30 .22700       34 6000.0      |
|15 RADIUS CAN CENTER HOLE (IN)    1.25 |16 CARBON POROSITY FACTOR   0.740 | CO  +  CO <-> C  + CO2  31 .70400E-01   35 25000.     |
|46 INIT CARBON LOADING   (LBM)    0.00 |42 MINI-TIMESTEP DAMP     0.80000 | C   + 2H2 <-> CH4       32 .10100E-06   36 .33000E-03 |
|12 MASS OF CATALYST      (LBM) 0.33000 |37 MAX MINI-TIMESTEP ITER   500.0 |10 REACTOR GAS DIFFUSION COEF        (FT^2/HR)   0.535 |
|23 COMP WASTE + WORK  (BTU/HR)  100.00 |43 REACTOR DAMPINMG       0.98000 |18 REACTOR GAS THERMAL G      (BTU/HR-FT^2-F)   0.020 |
|17 CARTRIDGE CAPACITY    (LBM)  20.000 |38 MAX RECTOR ITERATIONS    500.0 |44 REACT-CAB G (     )   0.20 45 CABIN TEMP (F)  72.000 |
|25 REACTOR SET POINT TEMP  (F)  1250.0 |39 RELAX CRITERION       0.10E-03 |22 CONDENS UA (     )  10.00 49 FUL DSPLY (N=0,Y=1) 0. |
|26 REACT SET POINT PRES (PSIA)  28.000 |7 POWER        (WATTS)   0.00E+00 |  HYDRAULIC EQUIV:  LENGTH 40   1.000   DIA 41   6.000 |
|28 INIT REACTOR PRESS   (PSIA)  14.700 |8 WEIGHT         (LBM)   0.00E+00 |--- BENCHMARK DATA -------- MIN ----- NOM ----- MAX ---|
|27 H2 MOLE FRACTION SET POINT   .44000 |9 VOLUME        (FT^3)   0.00E+00 | HEATER POWER REQ'D  (W) 0.000E+00 0.000E+00 0.000E+00 |
|_____________________________________| |_____________________________________| FEED STREAM IN  (LB/HR) 0.000E+00 0.000E+00 0.000E+00 |
|50 CO2 REDUCT RATE (LB/HR) 0.00E+00    51 TOTAL CO2 REDUCED (LB) 0.00E+00 | H2 MAKE-UP IN   (LB/HR) 0.000E+00 0.000E+00 0.000E+00 |
|55 CARB PROD RATE  (LB/HR) 0.00E+00    53 TOT CARB PRODUCED (LB) 0.00E+00 | CONDENSATE  OUT (LB/HR) 0.000E+00 0.000E+00 0.000E+00 |
|54 CONDENSATE FLOW (LB/HR) 0.00E+00    52 CURRENT CARB LOAD (LB) 0.00E+00 | CARB PROD RATE  (LB/HR) 0.000E+00 0.000E+00 0.000E+00 |
|57 H2 MAKE-UP RATE (LB/HR) 0.00E+00    56 AVG HEAT LOSS (BTU/HR) 0.00E+00 | BLEED STRM OUT  (LB/HR) 0.000E+00 0.000E+00 0.000E+00 |
|59 CARTRIDGE COUNT              0.0    58  * OPEN OUTPUT *       0.00E+00 | CO2 REDUCT RATE (LB/HR) 0.000E+00 0.000E+00 0.000E+00 |
|_____________________________________________________________________________|_____________________________________________________|
```

10.7 BUBBLE—Interconnect BUBBLE Simulation Description

The BUBBLE routine is a feature that facilitates the interconnection of flow streams on a different subsystem screen or to resolve an awkward interconnect within a single subsystem. The designer may use as many bubbles as is required to accomplish the definition of the system. The program will resolve the bubble interconnects prior to the solution of the model by connecting components that are attached to bubbles of the same name. This effectively eliminates BUBBLE's from the solution process.



10.7.1 General Discussion

BUBBLE's are eliminated prior to the program entering into the solution routines and, therefore, do not change the energy or mass content of the fluid. The primary restriction is that it may not be used to perform a split or sum operation except when connected to a STORE, CABIN, or SUM component.

10.7.2 Pressure Calculation

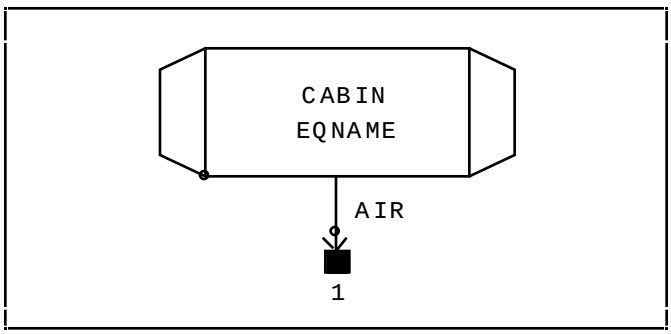
The logic used to resolve the BUBBLE interconnects will only utilize the equivalent diameter and length data for the last connection resolved. The user must either enter the identical equivalent length and diameter data for all connections to a given BUBBLE name or utilize the PIPE component to account for the pressure line loses.

10.7.3 Input Parameters

There are no user inputs.

10.8 CABIN—CABIN Simulation Description

This subroutine may be used to simulate a cabin or compartment. It is designed to track the respirable atmospheric composition, mass additions/losses, and heat transfer to and from other cabins, equipment, and the external orbital environment. The equipment loads are determined by assigning subsystems to a cabin through the "ASSIGN" command. The effective sink temperatures may be timelined through the use of the OPS block logic. The program is currently configured to simulate up to 25 cabins in any one test case.



10.8.1 General Description

The major assumptions used in the current formulation of the cabin routine include the following:

A. Interior mean radiant and structure temperatures are equal.

B. Mixing of constituents is instantaneous and uniform over the cabin volume.

C. Particle concentrations and temperatures exit the volume at the calculated perfectly mixed conditions.

D. Energy loss due to cabin leakage is negligible.

The following sections provide a brief technical discussion of the mass and energy balance relationships utilized in modeling this component.

10.8.2 Energy Balance

Determination of the bulk air temperature involves the application of the generalized conservation of energy equation (open system formulation) in which the kinetic and potential energy terms are neglected. The equation is expressed as follows:

$$\dot{Q} + \sum \dot{m}_i h_i = \sum \dot{m}_e h_e + \left.\frac{dE}{dt}\right|_{\text{cabin}} .$$

Implementing the above assumptions and considering air to behave like an ideal gas, the bulk air energy equation may be finite differenced into an algebraic equation. In its indexed form it is expressed as follows:

$$T_i^{P+1} = T_j + \frac{\sum G_{ij} T_j^P + Q_i}{\varepsilon \sum G_{ij} + C_i/\Delta t} , \qquad (1)$$

where,

T_i^{P+1} = implicitly calculated temperature of $i^{\underline{th}}$ cabin at end of time step, °F

T_i^P = temperature of $i^{\underline{th}}$ cabin at beginning of time step, °F

G_{ij} = conductance of adjoining nodes, i.e., $\dot{m}C_p$, $kA/\Delta X$, etc., Btu/h °F

C_i = capacitance of $i^{\underline{th}}$ cabin, Btu/h °F

Q_i = impressed heating rate on $i^{\underline{th}}$ cabin, Btu/h

Δt = time step, hours

$$\varepsilon = \frac{\Delta t - C_i/\sum G_{ij}\left(1 - e^{\frac{\Delta t}{C_i/\sum G_{ij}}}\right)}{\Delta t\left(1 - e^{\frac{\Delta t}{C_i/\sum G_{ij}}}\right)} .$$

Similar equations are utilized for other cabin nodes (i.e., structure, shell). A schematic of the cabin thermal network is shown in figure 55.

The cabin bulk air and structure nodes are thermal diffusion nodes. The thermal capacitance of the pressure shell, meteoroid shield, attachments, etc., is assumed to reside in the

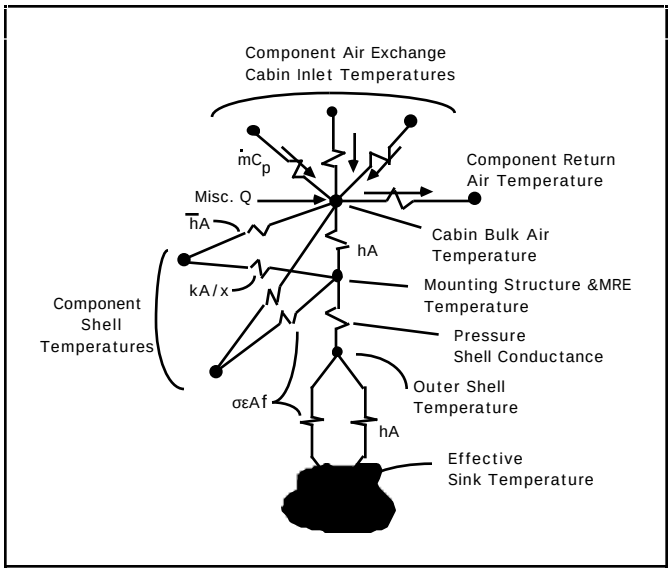


Figure 55. CABIN thermal network.

structure node. The cabin air mass, as calculated by the mass continuity equation, provides for the determination of the bulk air capacitance. The outer structure node is arithmetic, and the effective environmental sink temperature is considered to be a thermal boundary. Sink temperatures may be timelined as a function of orbital period and included in the supplemental OPS logic for a given specific case. Orbital heating loads may also be imposed on the cabin air directly through Q_i (CLOAD array) in supplemental OPS logic.

Equipment heat rejection to the cabin will occur via one or all of the following mechanisms. Flow enthalpy transfers to and from the cabin are iteratively transferred by the C and PRO arrays to the inlet/outlet streams of the cabin. Component convective loading on the bulk air node is communicated via the CLOAD array and is based on the delta temperatures that exist between the equipment shell and cabin air. Mechanical conduction is allowed to occur between the component and cabin structure node. Radiative exchange between equipment and cabin is also provided using a mean radiant environment approach. Heat transfer rates for radiation and conduction are based on the delta temperatures existing between the component shell and the solid structure node. These quantities are passed from the component routines to the cabin routine by the CLOAD array.

The implicit system of algebraic equations are solved by an iterative, block substitution technique. Internal to the routine a check is made on numerical integration stability based on the system time step and given conditions.

10.8.3 Mass Balance

The total atmospheric mass balance is performed by a flow balance on the participating individual species. The rate of constituent mass storage is governed by the conservation of mass equation and is given in its finite differenced (Euler) formulation as:

$$m_i^{P+1} = m_i^P + \Delta t \left(\sum \dot{m}_i^{\text{out}}\right) , \qquad (2)$$

and the total atmospheric mass is given by;

$$m_t = \sum_j m_j ,$$

where,

m_i^{P+1} = mass of $i^{\underline{th}}$ constituent at end of time step, lbm

m_i^P = mass of $i^{\underline{th}}$ constituent at beginning of time step, lbm

$\sum \dot{m}_i^{\text{out}}$ = flow of $i^{\underline{th}}$ constituent into and out of cabin, respectively, lbm/h.

Incrementing of the cabin air mass per the explicit formulation given above will occur after the system relaxation criteria has been achieved for the current time step. Numerical integration stability of the Euler formulation is dependent on the time step choice. For the most part, diminishing time steps will give correspondingly greater accuracy to the approximation (at the expense of run time). It is therefore left to the user to determine a suitable system time step.

10.8.4 Pressure Calculations

The cabin pressure is determined from the ideal gas laws and may vary over the given time step. The pressure is updated every iteration and includes the effects from cabin leakage, EVA, and experiment vent loses. Cabin relief venting is determined at the completion of the timestep in the 300 block.

10.8.5 Input Parameters

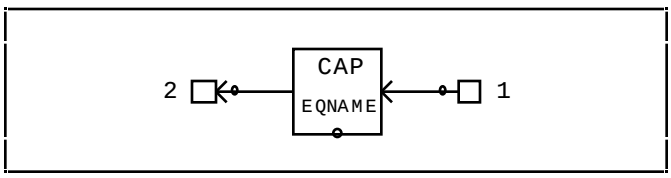
The input parameters are shown in table 26. The user must specify the cabin volume, structure mass and initial partial pressures, total pressure, and temperatures. An input is provided for the cabin pressure relief valve set point and for the experiment vent, EVA, and cabin leakage losses. These are specified as LBM/Day. Transient vent operations can be simulated through the OPS blocks by resetting the flowrates to the 24-h equivalent flowrate for the period in question. The cabin heat load input is used to simulate miscellaneous loads such as the cabin lighting, but can be used to simulate environmental loads if required. If environmental heat loads are being imposed through the cabin heat load input parameter, the heat transfer to the cabin sink should be adjusted accordingly. The interface to the external environment is simulated through the shell and external radiation conductances. The shell conductances should include any insulation present such as multilayer insulation (MLI). An external convection conductance is provided to simulate cabins in a terrestrial environment.

Table 26. CABIN edit screen.

```
CASE NAME: DEFAULT            CABIN EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                   CABIN NAME:                  RECORD NUMBER: 000001
_____________________________________________________________________________
|13 CABIN VOLUME      (FT^3)  2250.0 |28 STRUCTURE MASS          (LBM) 22000.0 |
|14 RELIEF SET POINT  (PSIA)  15.500 |20 CABIN HEAT LOAD       (WATTS)     0.0 |
|10 CABIN LEAKAGE   (LB/DAY)   5.000 |23 INTERIOR CONV      (BTU/HR/F)    23.2 |
|11 EXPERIMENT VENT (LB/DAY)   1.000 |24 SHELL CONDUCTANCE(BTU/HR/F)      19.4 |
|12 EVA AIR LOSS    (LB/DAY)   1.000 |22 ENVIRONMENT SINK TEMP     (F)   -75.0 |
|15 INITIAL TEMPERATURE  (F)  75.000 |29 EXTERNL CONVECTION (BTU/HR/F)    0.00 |
|16 INITIAL PRESSURE  (PSIA)  14.700 |25 EXTERNL RAD-K (BTU/HR/R^4)  0.171E-05 |
|21 INITIAL STRUCT TEMP  (F)  75.000 |                                         |
|17 INITIAL PPO2      (PSIA)   2.000 |  ** SUB-SOLUTION CONTROL PARAMETERS **  |
|18 INITIAL PPCO2    (MM HG)   3.000 |27 MAXIMUM ITERATIONS              200.0 |
|19 INITIAL REL HUM      (%)  40.000 |26 THERMAL RELAXATION COEFF      0.01000 |
|_____________________________________|_________________________________________|
|                ** BENCHMARK DATA **            |       ** OUTPUT DATA **       |
|                        MIN      NOM      MAX  |                               |
| TEMPERATURE      (F)   0.000    0.000    0.000|30 CABIN TEMP     (F)   0.000 |
| TOTAL PRESS   (PSIA)   0.000    0.000    0.000|31 TOTAL PRESS (PSIA)   0.000 |
| PPO2          (PSIA)   0.000    0.000    0.000|32 PPO2        (PSIA)   0.000 |
| PPCO2        (MM HG)   0.000    0.000    0.000|33 PPCO2       (PSIA)   0.000 |
| REL HUMIDITY     (%)   0.000    0.000    0.000|34 PPH2O       (PSIA)   0.000 |
| DEW POINT        (F)   0.000    0.000    0.000|                               |
|_____________________________________________|_______________________________|
```

10.9 CAP—Thermal Capacitor Simulation Description

This routine simulates the thermal behavior displayed by a phase-change thermal capacitor (CAP). It is modeled by means of a simple delta network/mass node arrangement. Isothermal charging/discharging operation will occur at the user-specified melt temperature. The total energy storage capacity is calculated as the product of the latent heat of fusion and capacitor mass. Calculated fluid outlet temperatures are assigned to stream 2 and will reflect both the isothermal and transient response characteristics of the process. The CAP does not generate flow; therefore, an active component must be upstream of the inlet. Pressure drop through the CAP is determined by a user-supplied equivalent line length and equivalent line diameter. The CAP data base parameters are shown in table 27.



Discussion

The CAP routine provides for the simulation of a phase change material (PCM) thermal capacitor. The capacitor is assumed to reside adjacent to an isothermal wall segment and is insulated from the environment on all other sides. The charge and discharge rates of the capacitor are determined via the heat transfer rates to and from the ATCS fluid loop. Simulation of convective heat transfer rates is provided using an arithmetic delta network. The network requires a user-specified effectiveness. Diffusional heat transfer rates between the wall segment and the PCM mass are computed using a variable conductivity. The conductance value is dependent upon the physical state of the capacitor mass (i.e., solid, liquid, or transition).

The composite thermal network is shown in figure 56. It consists of four nodes, distributed as follows: lumped capacitor mass (diffusion), isothermal wall segment and coldplate outlet (arithmetic), and coldplate inlet (boundary).

Several regimes of capacitor operation are possible. If inlet fluid temperatures are below the PCM melt temperature, two possibilities exist:

(1) The PCM is responding transiently with the ATCS fluid loop (sensible energy transfer)

(2) The PCM is discharging isothermally to the lower temperature sink.

If the inlet fluid temperatures are greater than the PCM melt temperature, two more possibilities exist:

(1) The PCM is charging isothermally (phase change energy storage)

(2) The capacitor is fully charged and responding transiently with the ATCS fluid loop (sensible energy transfer).

Isothermal charging/discharging of the capacitor will be achieved by assignment of a fictitiously large capacitance to the diffusion node. This will occur at the appropriate time, as determined by the constraints listed above.

Because of the introduction of the large capacitance, a mismanagement of energy is induced into the thermal network. It will be tracked internally and attributed to the phase change process. The maximum latent energy storage capacity is calculated as a function of the heat of fusion and PCM total mass.

Solution of the network equations is performed by a back substitution technique in which the diffusion nodal temperature is expressed in an implicit algebraic form. This allows for the iterative determination of all nodal temperatures as a function of capacitor state and fluid inlet temperature.

Pressure drop through the CAP is found by conventional pressure drop calculation methods utilizing a user-supplied equivalent length and equivalent diameter. See section 8.2 for a detailed explanation of the pressure drop calculations.

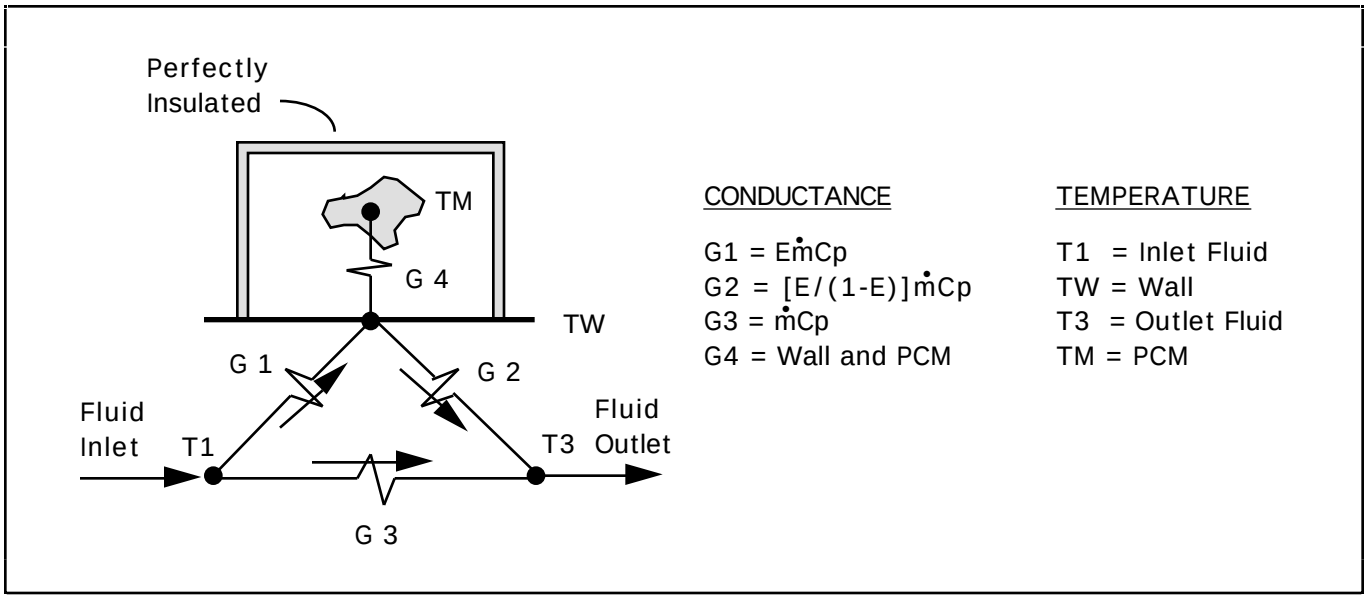


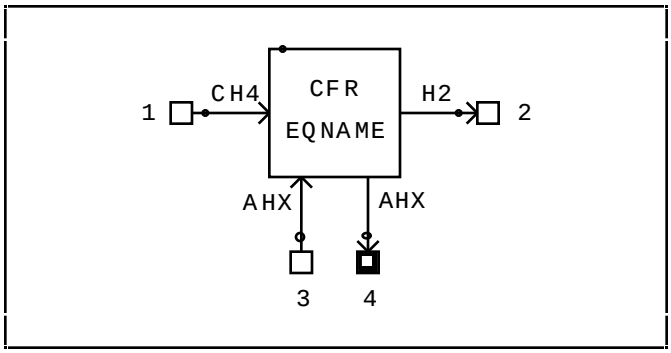
Figure 56. CAP thermal network.

Table 27. CAP edit screen.

```
CASE NAME: DEFAULT              CAP EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                     COMPONENT:                  RECORD NUMBER: 000001
_____________________________________________________________________________________
|        ** GENERAL INPUT DATA **          |                                        |
| 11 LUMPED CAP MASS    (LBM)       31.00 |       ** RESOURCE TRACKING **          |
| 12 LATENT HEAT OF FUS (BTU/LBM)  388.00 | 7 POWER    (WATTS)           0.00      |
| 13 PCM MELT TEMP      (F)        104.00 | 8 WEIGHT   (LBM)             0.00      |
| 22 INITIAL CAP TEMP   (F)         70.00 | 9 VOLUME   (FT^3)           0.000      |
| 23 INITIAL CP TEMP    (F)         70.00 |                                        |
|                                         |   ** PRESSURE DROP PARAMETERS **       |
|            ** THERMAL DATA **           | 20 EQUIVALENT LENGTH (FT)   10.00      |
| 14 SOLID PCM COND   (BTU/HR F)   741.34 | 21 EQUIVALENT DIA    (IN)   0.500      |
| 15 LIQUID PCM COND  (BTU/HR F)   741.34 |                                        |
| 16 TRANSITION PCM K (BTU/HR F)   741.34 |      ** SPECIFIC HEAT DATA **          |
| 17 FLUID TO WALL K  (BTU/HR F)    0.900 | 18 SOLID PCM  (BTU/LBM F)   0.500      |
|                                         | 19 LIQUID PCM (BTU/LBM F)   0.500      |
|                                         |                                        |
|_________________________________________|________________________________________|
|                 ** BENCHMARK DATA **                |     ** OUTPUT DATA **        |
|                     MIN        NOM        MAX      |24 HEAT STORE(KWH)    0.00    |
| TEMPERATURE (F)   0.00E+00   0.00E+00   0.00E+00   |25 CAP TEMP    (F)    0.00    |
| PRESSURE   (PSI)  0.00E+00   0.00E+00   0.00E+00   |26 CP TEMP     (F)    0.00    |
|_____________________________________________________|______________________________|
```

10.10 CFR—Carbon Formation Reactor Simulation Description

This component is used in conjunction with the Sabatier (SABAT) CO_2 reduction reactor, which converts carbon dioxide and hydrogen to methane and water. A condenser/separator removes the water from the SABAT outlet stream and the remaining CH_4 is fed into the CFR. The CFR decomposes the methane to solid carbon, which is stored for return to Earth, and hydrogen which is fed back into the Sabatier to close the cycle.



10.10.1 General Discussion

The CFR routine provides for the simulation of the heat and mass transfer involved in the methane pyrolysis process. The process chemical reaction is endothermic and occurs in the presence of a quartz catalyst as shown in the following stoichiometric equation:

$$CH_4 \underset{cat}{\rightarrow} C(s) + 2H_2 \ .$$

The following sections provide explanations of the mass and energy balance techniques used in the process model.

10.10.2 Mass Conversion Rate Determination

Mass conversion rates for the methane pyrolysis process are calculated as functions of the forward primary reaction stoichiometry and the inlet methane flow rate. Solid carbon generation is tracked internal to the routine. The routine output lists the current carbon loading as well as the number of carbon cylinder change outs at a given time in the simulation. All inert gases are passed through the CFR unaltered to maintain a mass balance.

10.10.3 Process Energy Balance

The CFR process consists primarily of a catalytic reactor, regenerative HX, and a noncondensing HX. A nodal thermal network of the model is shown in figure 57. The thermal vacuum jacket around the catalytic reactor provides insulation and reduces the power required to maintain the reactor at its 2,000 °F operating temperature. The regenerative HX (RHX) enables the hot outgoing H_2 stream to exchange its heat with the incoming CH_4 stream prior to entering the reactor. The RHX has an effectivity factor defined as follows:

$$\varepsilon_{RHX} = \frac{T_{hi} - T_{ho}}{T_{hi} - T_{ci}} \ ,$$

where,

T_{hi} = RHX hot side inlet

T_{ho} = RHX hot side outlet

T_{ci} = RHX cold side inlet

ε_{RHX} = RHX effectivity.

The H_2 outlet stream is cooled to ambient by circulating cooling air around the base of the reactor where the H_2 exits. This process is simulated using a four-pole noncondensing HX thermal network. The HX effectivity is calculated in the same manner as the RHX.

The catalytic reactor thermal behavior is simulated using “convective” delta and structure diffusion networks. The heater load is applied to the reactor mass node and is regulated to maintain the reactor at the user defined temperature set point (normally 2,000 °F). Heat transfer from the reactor to the vacuum jacket via radiation is simulated using standard linearization techniques for the determination of the radiation conductor. The vacuum jacket “touchable” outer surface is coupled to the cabin environment via user provided convection, conduction, and radiation conductors.

10.10.4 Numerical Solution

The CFR process equations determined from the thermal network are solved by an iterative

block substitution technique. Solution routine control parameters (relaxation and damping factors) are provided to ensure solution convergence for a wide range of boundary conditions.

10.10.5 Input Parameters

The user input parameters discussed previously for the CFR are shown in table 28.

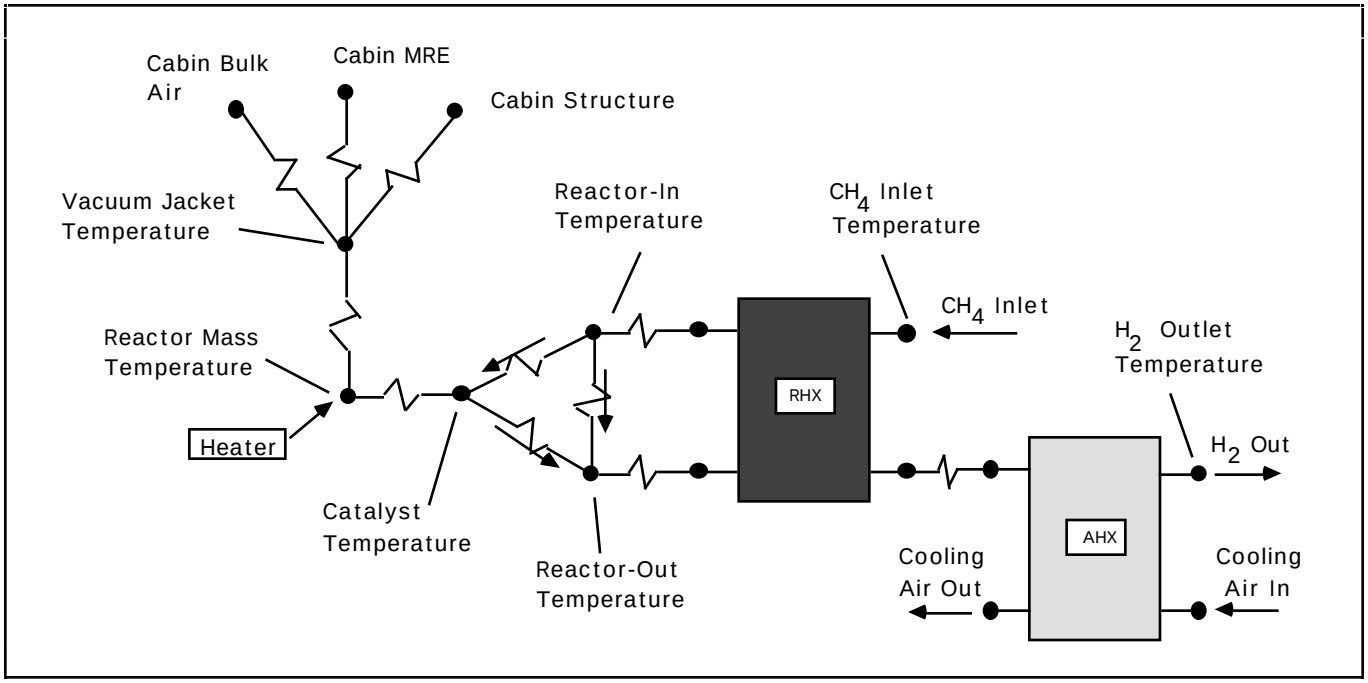


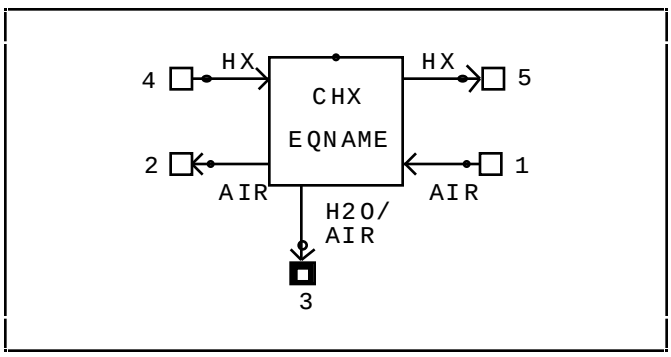
Figure 57. Carbon formation reactor thermal network.

Table 28. CFR edit screen.

```
CASE NAME: DEFAULT                                  CFR EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                 RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|        ** GENERAL INPUT DATA **        |            ** EFFECTIVITIES **            |                                          |
|21 REACTOR PRESSURE     (PSIA)   25.00  |18 RHX EFFECTIVITY                0.95000  |                                          |
|22 REACTOR SETPOINT TEMP   (F) 2000.00  |19 REACTOR EFFECTIVITY            0.99000  |        ** THERMAL INTERFACE DATA **      |
|23 THERMAL MASS        (BTU/F)   15.00  |20 HX EFFECTIVITY                 0.95000  |13 CONVECTION CONDUCTANCE   (BTU/HR/F)   0.100  |
|24 INITIAL MASS TEMP       (F) 1000.00  |        ** RESOURCE TRACKING **           |15 MOUNTING CONDUCTANCE     (BTU/HR/F)   0.250  |
|27 CARBON LOADING CAP    (LBM)   54.00  |7 POWER                  (WATTS)    0.00  |16 MASS TO CATALYST CONDUCT (BTU/HR/F)  25.000  |
|                                        |8 WEIGHT                   (LBM)    0.00  |14 RADIATIVE F*A*E              (FT^2)    0.05  |
|12 MAX HEATER POWER    (WATTS)  175.00  |9 VOLUME                  (FT^3)    0.00  |                                          |
|17 VAC JACKET F*A*E     (FT^2)    0.01  |                                          |    ** SUB-SOLUTION CONTROL PARAMETERS **  |
|                                        | ** COOLANT PRESSURE DROP PARAMETERS **   |25 MAXIMUM ITERATIONS                     200.0  |
|28 CONTROLLER GAIN                0.010 |10 EQUIVALENT LENGTH        (FT)    3.00  |26 DAMPING FACTOR (0 < X < 1)           0.50000  |
|29 N2 PURGE DURATION     (HRS)   12.00  |11 EQUIVALENT DIAMETER      (IN)    6.00  |30 TEMPERATURE RELAXATION          (F) 0.10000  |
|________________________________________|________________________________________|________________________________________|
|                                ** OUTPUT DATA **                                 | ** BENCHMARK DATA **  MIN        NOM         MAX   |
|33 TRANS CORE MASS TEMP   (F)     0.0 35 CH4 REDUCED/STEP       (LBM)  0.0000| CH4 IN    (LBM/HR) 0.000E+00  0.000E+00  0.000E+00 |
|34 CURRENT HEATR LOAD (WATTS)    0.00 36 CUMULATIVE CH4 REDUCED (LBM)    0.00| H2 OUT    (LBM/HR) 0.000E+00  0.000E+00  0.000E+00 |
|37 AVG HEATER POWER   (WATTS)    0.00 38 CURRENT CARBON LOADING (LBM)    0.00| C(S) GEN  (LBM/HR) 0.000E+00  0.000E+00  0.000E+00 |
|42 AVG ENV HEAT LOSS (BTU/HR)    0.00 39 TOTAL CARBON PRODUCED  (LBM)    0.00|________________________________________________|
|46 ENV ENERGY LOSS/STEP (BTU)    0.00 41 CARBON REMOVED/STEP    (LBM)  0.0000 44 TOT PURGED N2 (LBM)    0.00 47 N2 IN (LB)   0.00 |
|40 NUM CARTRIDGE CHANGEOUTS       0.0 43 REACTOR N2 LOADING     (LBM)    0.00 45 NUM OF N2 PURGES       0.00 48 TIME (HR)    0.00 |
|________________________________________________________________________________________________________________________________|
```

10.11 CHX—Condensing Heat Exchanger Simulation Description

This component is used to simulate the cabin humidity and temperature control. The component is modeled after the *Spacelab*/shuttle condensing HX. The overall condenser heat transfer coefficients, UA's, and condenser effectivities are internally calculated as functions of the primary and secondary inlet flows and specific heats. The UA data are entered as a second order curve fit of vendor data. Cabin air flows occur via the streams 1 and 2 and the condensed water and air exit through stream 3. The stream 3 is called the slurper and the flow rate is specified by the user. The routine is capable of sensible heat transfer alone if the inlet dew point is below the coolant temperature. The coolant flow side (secondary) is modeled as streams 4 and 5. Both latent and sensible heat load are removed by the coolant flow stream. The routine will notify the user of conditions where the inlet dew point is below the inlet coolant temperature.



10.11.1 General Assumptions

The CHX uses a wet/dry computation algorithm for providing the transient performance of a CHX. This technique divides the HX into two separate sections representing a sensible and a condensing region. The UA data are derived from vendor data, which are generally unique for the given configuration and design flow rates. The algorithm is based on film condensation. At present, only water vapor is condensed and no adjustments are made for competing condensables.

10.11.2 Mass and Energy Calculations

The moisture content of the gas outlet is a function of the outlet dew point and saturation pressures. The algorithm uses a two-step process that simulates the sensible section first and then the condensing section. If the inlet dry bulb temperature is not saturated; i.e., above the dew point, the first step is to reduce the temperature of the superheated gas to the dew point temperature. The temperature drop is a function of the air and coolant inlet temperatures and the HX effectivity (ε).

$$T_{ao} = T_{ai} - \varepsilon(T_{ai} - T_{wi}) \,, \tag{1}$$

$$\varepsilon = \frac{1 - e^{-(\mathrm{UA}(M_a)/\mathrm{M}\, C_{pa})\,(1 - M_r)}}{1 - M_r e^{-(\mathrm{UA}(M_a)/M\, C_{pa})\,(1 - M_r)}} \,, \tag{2}$$

where,

M = mass flow rate, lbm/h

Cp = specific heat, Btu/h °F

UA(M) = overall conductance as a function of mass flow rate, Btu/h °F

$M_r = M\, C_{pa}/M\, C_{pw}$.

In the second step, the moisture is removed in the condenser section. Through an iterative algorithm, outlet conditions are calculated that meet the requirement that the fraction of the dry HX UA used in the sensible section plus the fraction of the wet HX UA used in the condensing section is equal to 1.0. This assumes that the moisture is removed at constant dry bulb temperatures and the sensible heat is removed at a constant moisture content until the dew point is reached at the gas outlet temperature of the CHX. The required HX performance is calculated by the log mean temperature difference (LMTD) technique shown below:

$$\mathrm{LMTD}_s = \frac{((T_{ai} - T_{wo}) - (T_{dpi} - T_{wp}))}{LN((T_{ai} - T_{wo})/(T_{dpi} - T_{wp}))} \,,$$

$$\mathrm{LMTD}_c = \frac{((T_{ao} - T_{wi}) - (T_{dpi} - T_{wp}))}{LN((T_{ao} - T_{wi})/(T_{dpi} - T_{wp}))} \,.$$

The effective UA's are defined below as well as the sensible and latent fractions.

$$\mathrm{UA}_s = \mathrm{Q}_s/\mathrm{LMTD}_s \,,$$

$$\mathrm{UA}_c = \mathrm{Q}_c/\mathrm{LMTD}_c \,,$$

$$f_s = UA'_s/UA\ (M_a)\ ,$$

$$f_c = UA'_c/UA\ (M_a)\ ,$$

where,

Q_s = sensible load

Q_c = condenser load

f_s = sensible fraction

f_c = condenser fraction.

Convergence is generally attained in less than six iterations. A linear "zero error" extrapolation is performed after the first two loops to accelerate convergence.

10.11.3 Pressure Calculation

The pressure drop through the gas and coolant sides are based on equivalent length and diameter values specified by the user. The slurper outlet (stream 3) pressure is not dynamically determined, and the outlet pressure will be set by the down stream component if the user sets the pressure to 0.0 (default value) or will utilize the value specified by the user. If the user specifies a none zero value, the component assumes that there is an internal mechanism that controls the flow rate such as an internal pump.

10.11.4 Input Parameters

The nine input parameters and the output parameters are shown in table 29. The UA data are the result of a second-order curve fit of the vendor data. The UA data are in the form of

$$UA = C_1 + C_2 * (AIRFLOW) + C_3 * (AIRFLOW)^2\ ,$$

where the coefficients are entered through the input parameters shown below. The default data represents the vendor data for the *Spacelab* CHX.

References

1. "MDTSCO *Spacelab* Long Module ECLS Thermal Model Description and User Guide," MDC G9815, March 1981.

2. Humphries, W.R.: "A Method of Calculating Condensing Heat Exchanger Performance by Use of Steam Tables." NASA TM-53861, August 1, 1969.

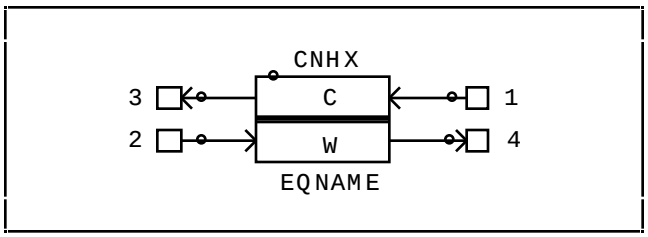
3. Randolph, W.O.: "Condensing Heat Exchanger Performance Calculation Technique." Memo for record, NASA, 1969.

Table 29. CHX edit screen.

```
CASE NAME: DEFAULT                                  CHX EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                                               COMPONENT:                                RECORD NUMBER:  000001
_____________________________________________________________________________________________________________________________
|          ** GENERAL INPUT DATA **          |    ** PRESSURE DROP PARAMETERS **    |       ** RESOURCE TRACKING **          |
|10 SLURPER AIR FLOWRATE  (LBM/HR)   35.00   |                                      |                                        |
|15 SLURPER OUTLET PRESSURE (PSIA)    0.00   |          * COOLANT SIDE *            |    7 POWER        (WATTS)   10.00      |
| POLYNOMIAL COEFFICIENTS FOR UA             |11 EQUIVALENT LENGTH  (FT)    1.00    |    8 WEIGHT         (LBM)   30.00      |
| UA(AIR)=C1+C2*VDOT+C3*VDOT^2+C4*VDOT^3     |12 EQUIVALENT DIA     (IN)  0.6250    |    9 VOLUME        (FT^3)    3.00      |
| 20 VDOT UNITS(0=LB/HR, 1=CFM,DEF    0.00   |            * AIR SIDE *              |                                        |
| S/L DEFAULTS: |16C1              222.000   |13 EQUIVALENT LENGTH  (FT)    1.00    |                                        |
| C1=1,C2->C4=0 | C2              0.855000   |14 EQUIVALENT DIA     (IN)  6.0000    |                                        |
| UPSIZED S/L: |18 C3          -0.118000E-03  |                                      |                                        |
| C1=2,C2->C4=0                0.000000E+00  |                                      |                                        |
|____________________________________________|______________________________________|________________________________________|
|                          ** OUTPUT DATA **                                      |                ** BENCHMARK DATA **                |
|    TEMPS (F)       INLET     OUTLET     CONDENSER LOADS (BTU/HR)                |                        MIN        NOM        MAX    |
|25,26 DEW POINT   41.7342    41.3907   27 LATENT          .956166 |CONDENSER LOAD (BTU/HR)  13952.60   16082.35   19136.90|
|29,30 COOLANT   40.          53.4374   28 SENSIBLE       15854.8 |CONDNSATE RATE (LBM/HR) .04862976  1.0149381  4.3953581|
|31,32 AIR SIDE    74.4432    41.5098   24 CUMUL H2O (LB) .39E+03 |AIR OUTLET TEMP     (F)     41.22      43.35      48.10|
|____________________________________________________________________|______________________________________________________________|
```

10.12 CNHX—Contact Heat Exchanger Simulation Description

This routine simulates the thermal control function of a CNHX. The routine assumes an isothermal control fluid as is normally encountered with two-phase flow. The outlet temperature is calculated utilizing a user-supplied HX effectiveness. The side of the CNHX with the two-phase fluid is considered the "control" side and the side with the single-phase fluid is considered the "working" side. The CNHX routine does not generate flow, therefore an active component must be upstream of each inlet. The pressure drop is calculated from user-supplied equivalent lengths and diameters. The CNHX data base parameters are shown in table 30.



Discussion

The CNHX routine assumes an isothermal control fluid (two-phase) and is modeled using a delta network scheme. The thermal network used for the CNHX is shown in figure 58. Later versions may incorporate stream enthalpy allowing the heat gain/loss of the two-phase fluid to be accounted for by the routine.

The CNHX fluid outlet temperature is calculated with the delta thermal network described in section 8.3.2. The CNHX requires only a user-input heat exchange effectiveness to utilize the delta network. This effectiveness can be a constant or could be calculated dynamically in the OPS code (if changing operating conditions warrant it) with an equation such as the following:

$$\varepsilon = 1 - \exp(hA/\dot{m}C_p) ,$$

where,

h = internal heat transfer coefficient, Btu/h/ft^2/°F

A = heat exchange area, ft^2

$\dot{m}C_p$ = fluid heat capacity rate, Btu/h/°F.

The internal heat transfer coefficient (h) would also have to be calculated in OPS code using conventional convection correlations for internal flow.

Pressure drop through the CNHX is found by conventional pressure drop calculation methods utilizing user-supplied equivalent lengths and equivalent diameters. See section 8.2.5 for a detailed explanation of the pressure drop calculations.

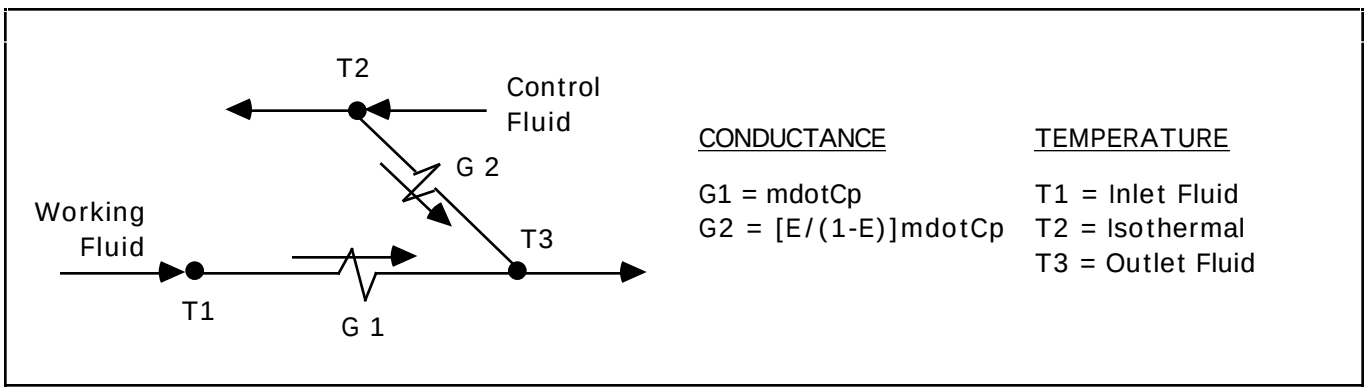


Figure 58. CNHX thermal network.

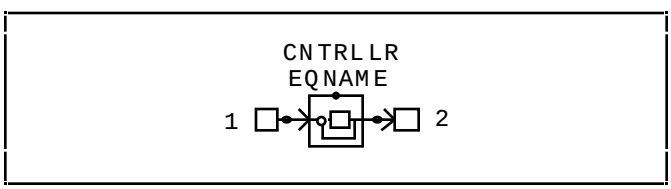
Table 30. CNHX edit screen.

```
CASE NAME: DEFAULT            CNHX EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                   COMPONENT:                  RECORD NUMBER: 000001
_____________________________________________________________________________
|                  ** INPUT DATA **                |  ** RESOURCE TRACKING **   |
|                                                  |                            |
| 10 EFFECTIVITY                           0.900   | 7 POWER  (WATTS)    0.00   |
|                                                  | 8 WEIGHT   (LBM)    0.00   |
|          ** PRESSURE DROP PARAMETERS **          | 9 VOLUME  (FT^3)   0.000   |
| 11 CONTROL SIDE EQUIVALENT LENGTH (FT)   10.00   |                            |
| 12 CONTROL SIDE EQUIVALENT DIA    (IN)   0.500   |                            |
| 13 WORKING SIDE EQUIVALENT LENGTH (FT)   10.00   |                            |
| 14 WORKING SIDE EQUIVALENT DIA    (IN)   0.500   |                            |
|_____________________________________________________|____________________________|
|                ** BENCHMARK DATA **              |      ** OUTPUT DATA **     |
|                     MIN        NOM        MAX    |                            |
| HEAT TRANS   (W)  0.00E+00   0.00E+00   0.00E+00| 15 HEAT TRANS  (W)    0.00 |
|_____________________________________________________|____________________________|
```

10.13 CNTRLLR—Controller Simulation Description

The routine CNTRLLR provides the user the capability to interactively monitor and control PRO, C, and CON array variables as well as certain system control variables such as TIME and STEP without resorting to OPS logic. OPS logic normally contained in OPS1 through OPS8 can be initiated from commands placed in the CNTRLLR CON array. Basic controller actions such as two position control (TPC) and proportional control (PC) are included as built-in options. Future CNTRLLR releases will contain canned options for various combinations of input and output derivative and integral compensation for the proportional control option. Alternatively, the CNTRLLR reverse polish notation programming language can be utilized to custom design control logic.

The CNTRLLR is placed on the subsystem screen with the LOcate command. Connections are made from the CNTRLLR component input and output hit boxes to the hit dots on the desired input/output streams or components. The hit dots on the CNTRLLR allow staging of CNTRLLR's where higher level CNTRLLR's control lower level ones. The arrows on the CNTRLLR streams denote the input and output CNTRLLR variables (i.e., a CNTRLLR connection to the inlet arrow signifies that the other end of that connection defines the component and/or stream of the input variable). The interconnecting control lines are drawn as dotted lines to provide distinction from the normal CASE/A stream connections. The CNTRLLR icon defining the respective streams is shown below.



10.13.1 General Discussion

User inputs to the CNTRLLR CON array include several system codes that control the timing and define the input and output variables to the CNTRLLR. These consist of an activation code, an input array code, an input item code, and an output item code:

Activation code	Similar to OPS logic location, this lets the user choose the location in the simulation where the CNTRLLR routine is activated. (1-Presimulation, 2-Before each time step, 3-Each iteration, 4-After each time step, 5-Output interval, 6-End of simulation, 7-Black Box)
Input array code	This code selects the array location of the CNTRLLR input variable (1-C array, 2-PRO array, 3-CON array).
Input item code	This code selects the location of the CNTRLLR input variable within the selected input array.
Output array code	This code selects the array location of the CNTRLLR output variable (1-C array, 2-PRO array, 3-CON array).
Output item code	This code selects the location of the CNTRLLR output variable within the selected output array.

In addition to these operational inputs the user must also provide the control logic to be performed. Several special CNTRLLR subroutines are available to the user in addition to a reverse polish programming language. A summary of the available commands and functions is provided in table 31. This presentation assumes the user is familiar with the reverse polish notation programming style and stack theory such as used in the Hewlett Packard programmable calculator. Where specific stack locations are referenced in the table, the notation stax(x) is used. When entering the program unit into the CNTRLLR data base a ":" must be used as a delimiter between arguments of a respective step.

The CNTRLLR is most useful when used as an alternative to OPS logic. In serving this function a CNTRLLR is required for each OPS block to be replaced. By using a CNTRLLR the user can avoid time consuming OPS logic compilations and linking required after changes in

logic, simply by changing the logic from within CASE/A and rerunning the case.

Debugging large cases can be simplified by using the CNTRLLR PRT command to print key variables interactively to the screen, enabling the user to monitor trends during the simulation.

To better depict the use of the CNTRLLR routine, the following example, based on the input data contained in table 31, is included.

10.13.2 CNTRLLR Example

This example illustrates the control of a coolant valve position by varying the pressure drop through the valve in terms of equivalent length (CNTRLLR output) based on a downstream fluid temperature (CNTRLLR input). The activation code is 4.0 indicating CNTRLLR activation after each time step. The input item is the second location of the PRO array (temperature) and the output item is the 11th item in the output component CON array (valve equiv length). The names of the input and output components are determined from the graphics as are the stream numbers. Step 1 and 2 call the canned proportional control algorithm with a setpoint of 115 ˚F, a gain of –30 in/˚F, a deadband of 1 ˚F, a maximum rate of change of 40,000 in/h, a minimum equivalent length of 0.5 in, and a maximum of 2,000 in. The position of the final control element (output) is returned to stax(1). The operator SOUT sets the CNTRLLR output to the current value of stax(1) in Step 3. Steps 4 and 5 print the valve equivalent length (LEQ1) and coldplate outlet temperature (TCP1) to the screen along with the specified labels. The END statement of Step 6 signals the end of the program unit.

10.13.3 Two Position Control Algorithm

The two position control algorithm will determine a final control element position (CNTRLLR output) based on a user specified setpoint, deadband and low and high output settings. If the measured input signal, INP is less than (XSET-DBAND/2) then the position of the final control element is returned as OUTLO to stax(1). If INP is greater than (XSET + DBAND/2) then the position of the final control element is returned as OUTHI to stax(1). CNTRLLR does not automatically set the output data location. To provide the greatest degree of flexibility this is left for the user to do with the SOUT command according to the output array and output item codes previously discussed. The format of this function is presented in table 32.

10.13.4 Proportional Control Algorithm

The proportional control algorithm will determine a final control element position (CNTRLLR output) based on a user specified setpoint, gain, deadband, maximum output rate, and maximum and minimum final control element position according to the following relationship:

$$P(n) = P(n-1)+\text{MGAIN}*(\text{INP-XSET}) ,$$

where,

$P(n)$ = position of final control element (CNTRLLR output)

$P(n-1)$ = previous position of final control element

MGAIN = main error signal gain factor

INP = controlled variable sensor signal

XSET = controlled variable setpoint command signal.

If the error signal (INP - XSET) is less than one-half the deadband then the control logic is skipped with no change in the position of the final control element. After calculation of P(n) by the above equation, P(n) is constrained by the user specified allowable rate and max/min positions before being placed in stax(1). CNTRLLR does not automatically set the output data location to the calculated value of P(n). This allows the user to modify the position of the final control element in the CNTRLLR program unit, if desired, before setting the specified output location. The format of this function is presented in table 34.

10.13.5 Planned Modifications

Future updates to the CNTRLLR will incorporate subroutines with more sophisticated control algorithms including input/output derivative/integral compensation. A version incorporating these capabilities now exists but has not been verified. Additionally planned modifications to the CNTRLLR data base will result in an indefinite number of available programming steps, and will provide the capability to insert/delete programming steps within the program unit.

Table 31. CNTRLLR data base parameters (example only).

```
CASE NAME: USL                                  CNTRLLR EDIT SCREEN                                          LAST UPDATED: 000000
SUBSYSTEM: TCS           COMPONENT:C1                                                                       RECORD NUMBER: 000001
 ____________________________________________________________________________________________________________________________
| ** INPUT DATA                                      INPUT ITEM CODE (ARRAY LOCATION)      2.0   |    POWER      (WATTS)   0.000   |
|       ACTIVATION CODE (0 < C < 7)       4.0        OUTPUT ARRAY CODE (1-C,2-P,3-K)       3.0   |    WEIGHT       (LBM)   0.000   |
|       INPUT ARRAY CODE  (1-C,2-P,3-K)   2.0        OUTPUT ITEM CODE (ARRAY LOCATION)    11.0   |    VOLUME      (FT^3)   0.000   |
|____________________________________________________________________________________________|_____________________________________|
| *PROGRAMMING STEPS *  9                  | 18 XXXXXXXXXXXXXXX | 27 XXXXXXXXXXXXXXX | 36 XXXXXXXXXXXXXXX | 44 XXXXXXXXXXXXXXX   |
| 1 PC:115:-30:1     | 10                  | 19 XXXXXXXXXXXXXXX | 28 XXXXXXXXXXXXXXX | 37 XXXXXXXXXXXXXXX | 45 XXXXXXXXXXXXXXX   |
| 2 C:40000:.5:2000  | 11                  | 20 XXXXXXXXXXXXXXX | 29 XXXXXXXXXXXXXXX | 38 XXXXXXXXXXXXXXX | 46 XXXXXXXXXXXXXXX   |
| 3 SOUT             | 12                  | 21 XXXXXXXXXXXXXXX | 30 XXXXXXXXXXXXXXX | 39 XXXXXXXXXXXXXXX | 47 XXXXXXXXXXXXXXX   |
| 4 PRT:LEQ1:OUT     | 13                  | 22 XXXXXXXXXXXXXXX | 31 XXXXXXXXXXXXXXX | 40 XXXXXXXXXXXXXXX | 48 XXXXXXXXXXXXXXX   |
| 5 PRT:TCP1:INP     | 14                  | 23 XXXXXXXXXXXXXXX | 32 XXXXXXXXXXXXXXX | 41 XXXXXXXXXXXXXXX | 49 XXXXXXXXXXXXXXX   |
| 6 END              | 15                  | 24 XXXXXXXXXXXXXXX | 33 XXXXXXXXXXXXXXX | 42 XXXXXXXXXXXXXXX | 50 XXXXXXXXXXXXXXX   |
| 7                  | 16 XXXXXXXXXXXXXXX | 25 XXXXXXXXXXXXXXX | 34 XXXXXXXXXXXXXXX | 43 XXXXXXXXXXXXXXX | ( ONLY STEPS 1-15    |
| 8                  | 17 XXXXXXXXXXXXXXX | 26 XXXXXXXXXXXXXXX | 35 XXXXXXXXXXXXXXX | 44 XXXXXXXXXXXXXXX |   CURRENTLY ACTIVE ) |
|=====================================================================================================================================|
| ** OUTPUT DATA                           | # OF FUNCTION STEPS     0. |     * SIGNAL VALUES *      |    * STORAGE REGISTERS *     |
|                                          | INPT CMPNT REL EQU #    0. | PRESENT INPUT    0.000E+00 | 1   0.000E+00 6   0.000E+00 |
| ** BENCHMARKS (SIGNAL VALUES)            | INPT CMPNT STRM #       0. | LAST INPUT       0.000E+00 | 2   0.000E+00 7   0.000E+00 |
|          MIN          NOM          MAX   | OUTPT CMPNT REL EQU #   0. | PRESENT OUTPUT   0.000E+00 | 3   0.000E+00 8   0.000E+00 |
|INP   0.000E+00    0.000E+00    0.000E+00 | OUTPT CMPNT STRM #      0. | LAST OUTPUT      0.000E+00 | 4   0.000E+00 9   0.000E+00 |
|OUT   0.000E+00    0.000E+00    0.000E+00 | TOTAL # OF LOOPS        0. |                            | 5   0.000E+00               |
|__________________________________________|____________________________|____________________________|____________________________|
```

Table 32. Available CNTRLLR commands.

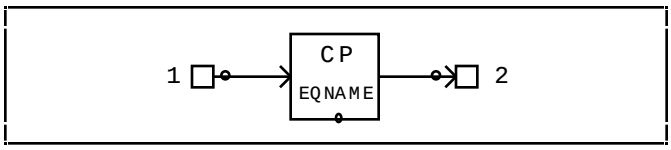
COMMAND	FUNCTION
+,-,*,/,**	Performs indicated mathematical operation on the current values of stax(1) and stax(2) placing the result in stax(1) and pushing the original stack values back one location.
CF (FC)	Converts the current value of stax(1) from Celsius (Fahrenheit) to Fahrenheit (Celsius).
DTIM	Retrieves the simulation time step and places it in stax(1).
E:VALUE	Places a specified VALUE in stax(1) as a real variable.
END	Program statement signaling the end of the program. Must be included as the last statement of the CNTRLLR program logic.
GK,GC,GP,GT SK,SC,SP,ST	Invokes the respective CASE/A subroutine to retrieve or set the current value of the specified data location to the current value of stax(1). The format of this command is: XK:NAME:LOCATION XC:NAME:STREAM:CONSTITUENT XT:NAME:STREAM XP:NAME:STREAM where NAME, STREAM, LOCATION, and CONSTITUENT are the same as for the CASE/A GET (SET) routines discussed in section 7.
GOTO:STEP#	Transfers control to the specified step number. STEP# must be an integer value ranging from 1 to the maximum number of steps--currently 15.
IFEQ,IFLT, (IFXX:VALUE)	Skips the next program step if the current value of stax(1) IFLE,IFGT, satisfies the logical test as compared to the real variable IFGE,IFNEVALUE.

Table 32. Available CNTRLLR commands (continued).

INP (OUT)	Places the current value of the input (output) signal in stax(1).
LINP (LOUT)	Places the previous value of the input (output) signal in stax(1).
M:REG#	Recalls a real value from the specified storage register and places it in stax(1). REG# must be an integer variable ranging from 0 through 9.
PC	Calls the basic proportional control algorithm and returns a calculated output value to stax(1). A discussion of the proportional control algorithm is included in the text. (Format- PC:XSET:MGAIN:DBAND C:RMAX:OUTLO:OUTHI): MGAIN = Main error signal gain factor INP = Controlled variable sensor signal XSET = Controlled variable setpoint command signal DBAND = Deadband C = Continuation line specifier RMAX = Maximum rate of output change OUTLO = Minimum allowed output value OUTHI = Maximum allowed output value
PRT:ALP:PVAR	Interactively prints to the screen an alphanumeric message (ALP) up to eight characters long and the current value of a specified print variable (PVAR). Allowable print variables are as follows: SX1-SX9 = Stax locations 1-9 SR1-SR9 = Storage locations 1-9 INP = Current input value LINP = Previous input value OUT = Current output value LOUT = Last output value X = Null space
SIN,COS,TAN, ASIN,ACOS,ATAN, SINH, COSH,TANH, EXP ,LN, LOG, SQRT, 1/X	Performs indicated mathematical operation on the value of stax(1), placing the result in stax(1) and pushing the original .stax(1) to stax(2).
SM:REG#	Stores the current value of stax(1) in the specified storage register. REG# must be an integer variable ranging from 0 through 9.
SOUT	Sets the value of the CNTRLLR output data location to the current stax(1) value.
TIM	Retrieves the current simulation time and places it in stax(1).
TPC	Calls the basic two position control algorithm and returns a calculated output variable to stax(1). A discussion of the two position control algorithm is included in the text. (Format- TPC:XSET:OUTLO:OUTHIC:DBAND) where; XSET = Controlled variable setpoint OUTLO = Output setting corresponding to a low input signal OUTHI = Output setting corresponding to a high input signal C = Continuation line specifier DBAND = Control deadband
X-Y	Exchanges the current value of stax(1) and stax(2).

10.14 CP—Coldplate Simulation Description

This routine simulates the operational and thermal characteristics of a typical electronic device mounted on a coldplate (CP) assembly. The electrical heat load is specified by the user. Baseplate, equipment, and shell temperatures are tracked by the routine. The equipment mass will exhibit transient thermal behavior. The CP component allows for heat exchange with its surroundings. The CP outlet fluid temperature is calculated with a delta network scheme. The CP does not generate flow, therefore an active component must be upstream of the inlet. Pressure drop through the CP is determined by a user-supplied equivalent line length and equivalent line diameter. The CP data base parameters are shown in table 33.



Discussion

The CP utilizes the thermal network described in section 8.3.3 and its adaptation for the CP is shown schematically in figure 59. The shell and wall have no mass (arithmetic nodes) while the equipment has a thermal capacitance (diffusion node). This thermal network allows heat exchange with the surroundings through convective, conductive, and radiative pathways in conjunction with heat transfer between the equipment mass and its shell and baseplate. These heat transfer paths can be modified by the user with normal component data base editing procedures. It should be noted that the CP can simulate a perfectly insulated shell by setting the mass-to-shell thermal conductor to zero. The heat load is applied directly to the equipment mass and can be timelined with OPS code if desired.

The CP fluid outlet temperature is calculated with the delta thermal network described in section 8.3.2. The CP requires only a user-input effectivity to utilize the delta network. This effectivity can be a constant or could be calculated dynamically in the OPS code (if changing operating conditions warrant it) with an equation such as the following:

$$\varepsilon = 1 - \exp(hA/\dot{m}C_p) ,$$

where,

h = internal heat transfer coefficient, Btu/h/ft^2/F

A = heat exchange area, ft^2

$\dot{m}C_p$ = fluid heat capacity rate, Btu/h/°F.

The internal heat transfer coefficient (h) would also have to be calculated in OPS code using conventional convection correlations for internal flow.

Pressure drop through the CP is found by conventional pressure drop calculation methods utilizing a user-supplied equivalent length and equivalent diameter. See section 8.2.5 for a detailed explanation of the pressure drop calculations.

Table 33. CP edit screen.

```
CASE NAME: DEFAULT                CP EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                      COMPONENT:                 RECORD NUMBER: 000001

            ** GENERAL INPUT DATA **          |     ** RESOURCE TRACKING **
 10 THERMAL LOAD          (WATTS)   500.00    |
 11 EFFECTIVENESS                     0.90    | 7 POWER           (WATTS)    0.00
                                              | 8 WEIGHT            (LBM)    0.00
 13 INITIAL MASS TEMP         (F)    70.00    | 9 VOLUME           (FT^3)    0.00
 16 THERMAL MASS          (BTU/F)    10.40    |
                                              |   ** THERMAL INTERFACE DATA **
      ** PRESSURE DROP PARAMETERS **          | 19 SHEL-ENV HA (BTU/HR/F)    0.00
 14 EQUIVALENT LENGTH        (FT)   10.000    | 20 SHEL-ENV FAE    (FT^2)    0.00
 15 EQUIVALENT DIAMETER      (IN)    0.500    | 21 SHL-ENV KA/X  (B/HR/F)    0.00
                                              | 22 MAS-SHL KA/X  (B/HR/F)    1.00
       ** THERMAL CONDUCTANCE DATA **         |
 17 CONTACT AREA           (FT^2)   2.0000    | 23 THERM RELAX CRITERIA .1000E-04
 18 CONTACT COND (BTU/HR/FT^2/F)   20.000     | 24 THERM MAX LOOP           100.0

                 ** BENCHMARK DATA **                |     ** OUTPUT DATA **
                      MIN        NOM        MAX      |
 BASEPLT TEMP (F)  0.00E+00   0.00E+00   0.00E+00 | 27 BASEPL TMP (F) .00E+00
 EQUIPMT TEMP (F)  0.00E+00   0.00E+00   0.00E+00 | 28 EQUIP TEMP (F) .00E+00
```

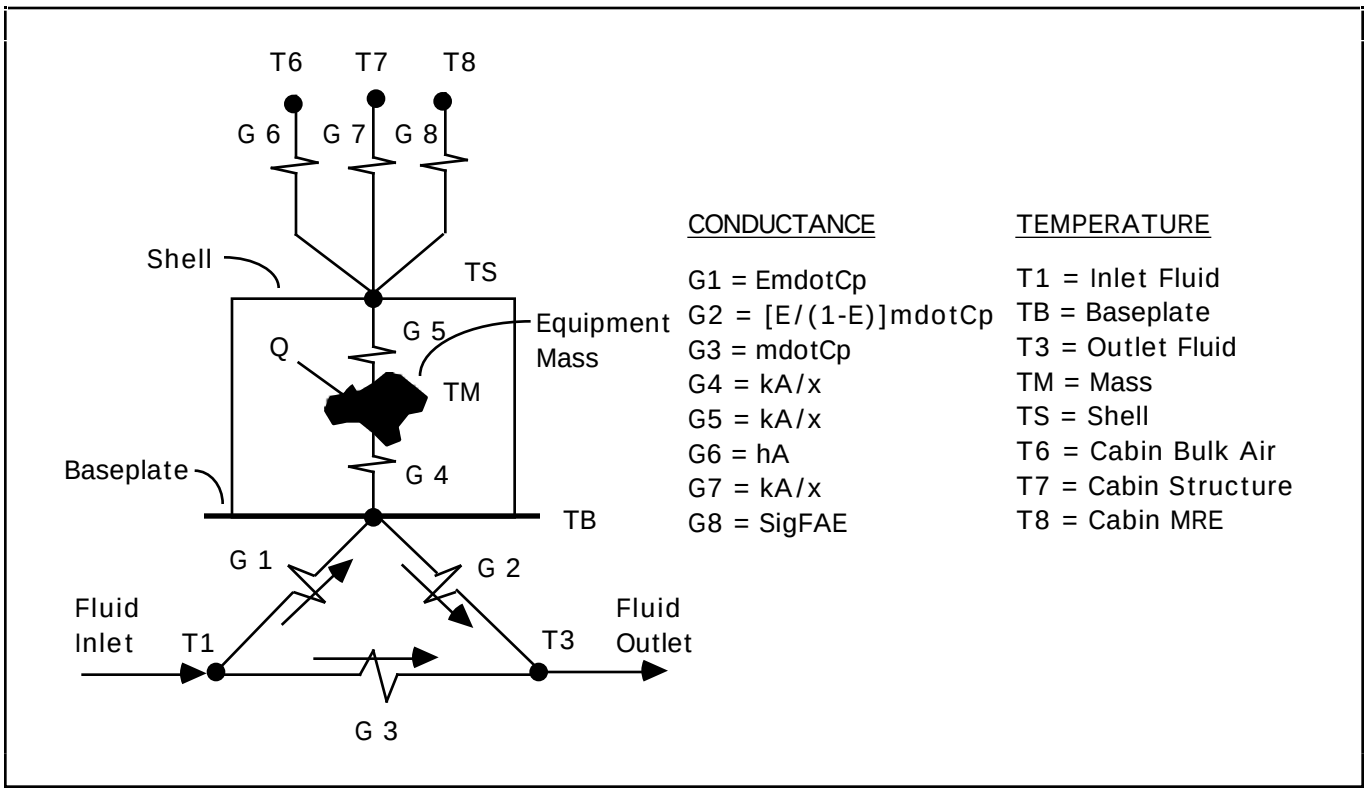
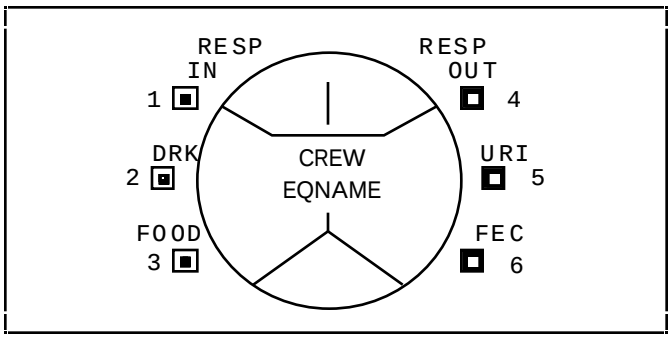


Figure 59. CP thermal network.

10.15 CREW—CREW Simulation Description

The CREW routine provides the metabolic forcing functions required to drive the ECLSS consumables/thermal simulation. The user inputs are the crew size and the unit consumption and production rates of the principal metabolic products. The outlet stream compositions are also user defined.



10.15.1 General Discussion

This routine bases overall metabolic production or consumption rates on user input data values which are read into the program on a pounds per man day basis. This allows the user to input production or consumption rates of the following principal metabolic items: (a) number of crew members, (b) metabolic O_2 demand, (c) metabolic respiration and perspiration output, (d) drinking H_2O demand, (e) amount of H_2O required for food preparation, (f) consumption of dry food, (g) urine production, (h) feces production, and (i) crew member sensible heat load. The compositions of the urine and feces output streams are defined by the user through the input of the individual constituent mass fractions.

The routine pulls the required metabolic needs from the current composition of entering streams 1 (breathables), 2 (water), and 3 (food). Production of metabolic materials are released to the cabin or tanks through streams 4 (breathables + CO_2 + perspiration), 5 (urine), and 6 (feces). Total crew sensible heat loads are impressed on the cabin environment once per time step. The number of crew members can be set to fractional values to simulate animals or may be set to zero.

10.15.2 Input Parameters

The user input parameters for the CREW routine are shown in table 34. The constituents data base is shown in table 35. Normally, the user need only provide the number of crew persons. The default consumption and production rates are based on average human intake and output data and should not be changed for normal simulations.

Table 34. CREW edit screen.

```
CASE NAME: DEFAULT            CREW EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                      RECORD NUMBER: 000001
_____________________________________________________________________________
|                                 ** INPUT DATA **                            |
||10 NUMBER OF CREW PERSONS            8.0 |16 URINE OUTPUT(LB/MN/DAY)    3.30 |
||11 METABOLIC O2 DEMAND(LB/MN/DAY)    1.84 |17 FECES OUTPUT(LB/MN/DAY)    1.00 |
||12 DRINK H2O DEMAND   (LB/MN/DAY)    4.09 |                                   |
||13 FOOD PREP H2O      (LB/MN/DAY)    8.00 |     ** RESOURCE TRACKING **       |
||14 FOOD DEMAND        (LB/MN/DAY)    1.00 | 7 POWER           (WATTS)    0.00 |
||18 SENSIBLE HEAT LOAD(BTU/MN/DAY) 9420.00 | 8 WEIGHT            (LBM)    0.00 |
||                                          | 9 VOLUME           (FT^3)    0.00 |
|| CONSTITUENT EMPIRICAL DATA (^L TO OPEN)_ |                                   |
||  RESPN & PERSPN OUTPUT  ITEMS 23 TO 70   |                                   |
||  URINE OUTPUT                 72 TO 120  |                                   |
||  FECAL OUTPUT                122 TO 170  |                                   |
||_________________________________________|_________________________________|
```

Table 35. CREW constituent edit screen.

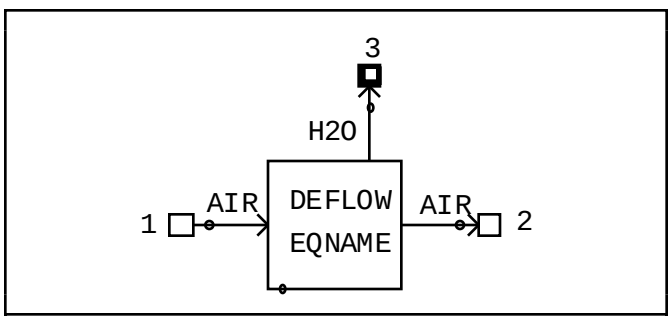
CASE NAME: DEFAULT — CREW CONSTITUENT SCREEN — LAST UPDATED: 890215

SUBSYSTEM: DEFAULT COMPONENT: — RECORD NUMBER: 000001

LABEL	R&P OUTP LB/MN-DY	URINE MS FRACTION	FECES MS FRACTION	LABEL	R&P OUTP LB/MN-DY	URINE MS FRACTION	FECES MS FRACTION	LABEL	R&P OUTP LB/MN-DY	URINE MS FRACTION	FECES MS FRACTION
CONSTIT 2	xxxxxxxx	0.0000000	0.0000000	CONSTIT 19	0.00000	0.0000000	0.0000000	CONSTIT 36	0.00000	0.0000000	0.0000000
CONSTIT 3	0.00000	0.0000000	0.0000000	CONSTIT 20	0.00000	0.0000000	0.0000000	CONSTIT 37	0.00000	0.0000000	0.0000000
CONSTIT 4	2.20000	0.0000000	0.0000000	CONSTIT 21	0.00000	0.0000000	0.0000000	CONSTIT 38	0.00000	0.0000000	0.0000000
CONSTIT 5	0.00000	0.0000000	0.0000000	CONSTIT 22	0.00000	0.0000000	0.0000000	CONSTIT 39	0.00000	0.0000000	0.0000000
CONSTIT 6	0.00000	0.0000000	0.0000000	CONSTIT 23	0.00000	0.0000000	0.0000000	CONSTIT 40	0.00000	0.0000000	0.0000000
CONSTIT 7	0.00000	0.0000000	0.0000000	CONSTIT 24	0.00000	0.0000000	0.0000000	CONSTIT 41	0.00000	0.0000000	0.0000000
CONSTIT 8	4.02000	1.0000000	1.0000000	CONSTIT 25	0.00000	0.0000000	0.0000000	CONSTIT 42	0.00000	0.0000000	0.0000000
CONSTIT 9	0.00000	0.0000000	0.0000000	CONSTIT 26	0.00000	0.0000000	0.0000000	CONSTIT 43	0.00000	0.0000000	0.0000000
CONSTIT 10	0.00000	0.0000000	0.0000000	CONSTIT 27	0.00000	0.0000000	0.0000000	CONSTIT 44	0.00000	0.0000000	0.0000000
CONSTIT 11	0.00000	0.0000000	0.0000000	CONSTIT 28	0.00000	0.0000000	0.0000000	CONSTIT 45	0.00000	0.0000000	0.0000000
CONSTIT 12	0.00000	0.0000000	0.0000000	CONSTIT 29	0.00000	0.0000000	0.0000000	CONSTIT 46	0.00000	0.0000000	0.0000000
CONSTIT 13	0.00000	0.0000000	0.0000000	CONSTIT 30	0.00000	0.0000000	0.0000000	CONSTIT 47	0.00000	0.0000000	0.0000000
CONSTIT 14	0.00000	0.0000000	0.0000000	CONSTIT 31	0.00000	0.0000000	0.0000000	CONSTIT 48	0.00000	0.0000000	0.0000000
CONSTIT 15	0.00000	0.0000000	0.0000000	CONSTIT 32	0.00000	0.0000000	0.0000000	CONSTIT 49	0.00000	0.0000000	0.0000000
CONSTIT 16	0.00000	0.0000000	0.0000000	CONSTIT 33	0.00000	0.0000000	0.0000000	CONSTIT 50	0.00000	0.0000000	0.0000000
CONSTIT 17	0.00000	0.0000000	0.0000000	CONSTIT 34	0.00000	0.0000000	0.0000000				
CONSTIT 18	0.00000	0.0000000	0.0000000	CONSTIT 35	0.00000	0.0000000	0.0000000				

10.16 DEFLOW—Dehumidifier (Flow Desorb) Simulation Description

The flow desorb desiccant bed (DEFLOW) routine simulates the transient water vapor removal characteristics of a molecular sieve containing up to two of the following materials: 10X Zeolite, 13X Zeolite, and silica gel. Detailed two-bed simulation requires the system time step to be less than the individual cyclic intervals. OPS logic is required to properly cycle the air flow to the proper components. Process cabin air is passed through the assembly via streams 1 and 2 where the water vapor mass transfer rate is calculated as a function of equilibrium isotherm data (i.e., bed loading and transient packed bed temperature). During the regeneration cycle (i.e., desorption cycle) of the desiccant bed, the flow direction is reversed using appropriate external valving. CASE/A does not currently support true reverse flow. Therefore, the DEFLOW is corrected for one-way flow, and flow reversal is handled internally to the component routine. The vacuum desorption mode has not been verified and, therefore, is currently disabled. The experienced user may provide any necessary code should this option be desired for a particular model. If it is not used, a "dummy" storage tank (or sink) should be connected to stream 3.



10.16.1 General Assumptions

The following general assumptions were made in developing the DEFLOW component routines.

A. Bed characteristics are uniform over a cross-section normal to the flow direction (excluding intraparticle concentration gradients).

B. Ideal plug flow exists, i.e., no channeling of the flow.

C. Adsorption occurs by diffusion of the adsorbate through the stagnate surface film on the surface of an adsorbent particle and then diffusion into the interior of the particle.

D. Heats of adsorption are independent of bed temperature and coverage.

E. Temperature gradients within an adsorbent pellet are negligible.

F. Water poisoning effects are ignored.

10.16.2 Mass and Energy Balance

The analytical approach to modeling the process for mass and heat transfer is based on the R-C network technique discussed in the molecular sieve routine description (see section 10.28). The DEFLOW component differs from the MOLSIEV component in that it is a passive, flow desorbed device. The desorbing water vapor leaves the desiccant bed by mechanisms identical to the adsorb cycle. The driving force is the difference between the water vapor density on the adsorbent pellet surface and the air free stream. The user must specify flow direction (stream 1 to stream 2 or vice versa) by setting the appropriate CON array location from OPS logic.

10.16.3 Input Parameters

The input parameters for DEFLOW are presented in table 36. The default data have been derived from published literature when possible. Any unavailable data were determined through parametric experimentation or estimated as necessary. Vacuum desorption parameters are provided in table 37 should the user activate this mode of the component. Outputs and benchmarks are listed in table 38.

References

1. "Verification Analysis of the Computer Aided System Engineering and Analysis (CASE/A) Molecular Sieve (MOLSIEV) and Desiccant Bed (DEFLOW) Component Routines," MDC W5183.

2. "Computer Analysis Techniques for Spacecraft Adsorption Beds," A69-21931.

3. Benet and Myers: “Momentum, Heat, and Mass Transfer.” McGraw-Hill Book Company, 1954, pp 509–511.

4. McAdams: “Heat Transmission.” Third edition, McGraw-Hill Book Company, 1962, pp 294–296.

5. “Carbon Dioxide Adsorption Isotherm Data Sheet No. 22, Molecular Sieve Type 5A Pellets,” Union Carbide.

6. “A Transient Performance Method for CO_2 Removal With Regenerable Adsorbents,” NASA CR-112098.

Table 36. DEFLOW edit screen.

```
CASE NAME: DEFLOW                                   DEFLOW EDIT SCREEN                                  LAST UPDATED: 890828
SUBSYSTEM: DEFAULT       COMPONENT: VALUES                                                            RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|              ** GENERAL INPUT DATA **                    |   ** DESCRIPTION OF BED MATERIALS  **          -1-            -2-      |
|    10 INITIAL BED LOAD, LB-H2O          -1.00            | MATERIAL TYPE   (0=13X,1=SILGEL,2=10X)  25      1.      26      0.     |
|    11 INIT BED/CAN TEMP, F               75.0            | NUMBER OF ELEMENTS (TOTAL <= 20)        12      4.      13      2.     |
|    44 CANISTER WEIGHT,  LBM              13.5            | MATERIAL WEIGHT LB                      27   9.20       28   9.10      |
|    15 OP MODE: 0=ADSORB, 1=DESORB            0.          | BULK SURFACE AREA, FT^2/LB              35   12.0        36   10.0      |
|    63  ** OPEN INPUT **                0.000E+00         | HEAT OF ADSORPTION, BTU/LBM-H2O         33  0.180E+04   34  0.180E+04 |
|    64  ** OPEN INPUT **                0.000E+00         | H2O MASS TRANSFER COEFFICIENT FT/HR     29  0.100E+04   30  0.100E+04 |
|    65  ** OPEN INPUT **                0.000E+00         | H2O INTRA-PARTICLE MASS G, FT^3/HR      31  0.100E+04   32  0.100E+04 |
|    66  ** OPEN INPUT **                0.000E+00         | INTER-ELEMENT MASS G, FT^3/HR           16  0.000E+00                  |
|                                                          |                                                                        |
|                ** SOLUTION CONTROL **                    | **ENVIRONMENTAL THERMAL CONDUCTANCES    BED/CANISTER     VACUUM PIPE   |
|    40 RELAXATION CRITERION             0.100E-04         |      SHELL-ENV CONDUCTION  BTU/HR/F     22   1.00        37   20.0      |
|    43 MAX THERMAL ITERATION COUNT       50.0             |      SHELL-ENV CONVECTION  BTU/HR/F     23   3.00        38   2.50      |
|    41 DAMPING FACTOR (0<D<1)           0.500             |      SHELL-ENV RAD F*A*E   FT^2         24   2.00        39   7.50      |
|                                                          |      CANISTER INSULATION G, BTU/HR/F    20   1.00                      |
| **PRESS DROP (AIR STRMS)**    07 POWER WATTS   0.000E+00 |      BED-CANISTER CONDUCT,  BTU/HR/F    21   7.00                      |
| 45 EQUIV LEN FT   1.00        08 WEIGHT LBM    0.000E+00 |      INTER-ELEMENT THERM G, BTU/HR/F    19  0.000E+00                  |
| 46 EQUIV DIA IN   6.00        09 VOLUME  FT^3  0.000E+00 |      FREE STRM COEFF,  BTU/FT2/HR/F     14   15.0                      |
|                                                          |      VAC PIPE THERM CAPAC,  BTU/LB-F                    58  0.100      |
| ^L TO VAC DESORB PARAMS  _   ^L TO OUTPUTS/BENCHMARKS  _ |      VAC PIPE THERM EFFECTIVITY                         59  0.990      |
|__________________________________________________________|________________________________________________________________________|
```

Table 37. DEFLOW vacuum desorb edit screen.

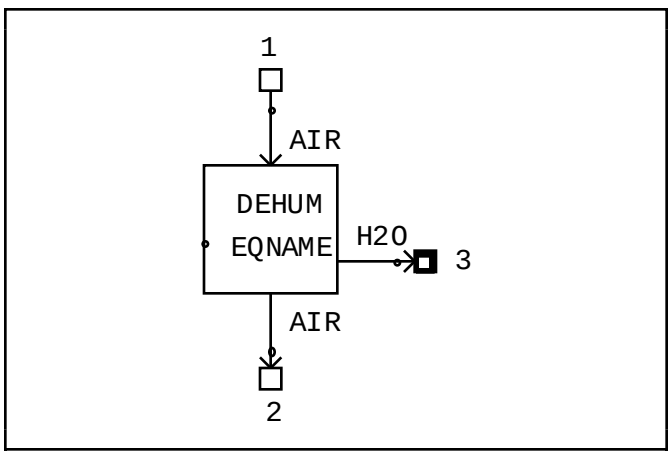
```
CASE NAME: DEFLOW              VACUUM DESORB EDIT SCREEN     LAST UPDATED: 890828
SUBSYSTEM: DEFAULT                COMPONENT: VALUES          RECORD NUMBER: 000001
 ____________________________________ ____________________________________________
|    **  HEATER CONTROL DATA  **     |        **  VACUUM DESORB DATA  **          |
| 47 HEATER FLAG ( 0=OFF,1=ON)     0. | 51 VAC FLAG (0=FLOW,1=PUMP,2=VENT)  0.     |
| 48 MAX HEATER PWR, WATTS  0.100E+04 | 52 VAX PUMP DISP, FT^3         115.        |
| 49 SET POINT TEMP, F        400.   | 53 CLEARANCE FACTOR          0.200E-01     |
| 50 HEATER CONTRLLR GAIN   0.100    | 54 POLYTROPIC EXPONENT        1.11         |
| 68 ** OPEN INPUT **       0.000E+00 | 55 VACUUM PIPE LENGTH, FT    0.000E+00     |
| 69 ** OPEN INPUT **       0.000E+00 | 56 VACUUM PIPE DIAM,   IN    0.000E+00     |
|                                     | 57 VAC PIP OUTLET PRESS,PSIA 0.100E-06     |
|                                     | 60 VAC SWITCH (0=OFF,1=ON          0.      |
|                                     | 61 ** OPEN INPUT **          0.000E+00     |
|                                     | 62 ** OPEN INPUT **          0.000E+00     |
|_____________________________________|____________________________________________|
```

Table 38. DEFLOW outputs/benchmarks edit screen.

```
CASE NAME: DEFLOW              DEFLOW OUTPUTS EDIT SCREEN    LAST UPDATED: 890828
SUBSYSTEM: DEFAULT                COMPONENT: VALUES          RECORD NUMBER: 000001
 ____________________________ ____________________________________________________
|                              *** OUTPUT DATA ***                                |
|                                                                                 |
|             60 CURRENT H2O BED LOADING,          LBM   0.000E+00               |
|             61 CURRENT ADSORPTION RATE,        LBM/HR   0.000E+00               |
|             62 CURRENT DESORPTION RATE,        LBM/HR   0.000E+00               |
|             63 CURRENT HEATER POWER (0=OFF),    WATTS   0.000E+00               |
|             64 AVERAGE BED PRESSURE,PPH2O       MM-HG   0.000E+00               |
|             65 *** OPEN OUTPUT  ***                     0.000E+00               |
|_________________________________________________________________________________|
|                             *** BENCHMARK DATA ***                              |
|                                                                                 |
|                                 MIN           NOM           MAX                 |
|        AIR IN   (LBM/HR)      0.000E+00     0.000E+00     0.000E+00             |
|        AIR OUT  (LBM/HR)      0.000E+00     0.000E+00     0.000E+00             |
|        H2O OUT  (LBM/HR)      0.000E+00     0.000E+00     0.000E+00             |
|        ** OPEN BENCH **       0.000E+00     0.000E+00     0.000E+00             |
|_________________________________________________________________________________|
```

10.17 DEHUM—Dehumidifier (Vacuum Desorb) Simulation Description

The dehumidifier (DEHUM) routine simulates the transient water vapor removal characteristics of a Zeolite (10X, 13X) or silica gel molecular sieve. The routine will time average process flow rates and cabin heat loading if the time step is not compatible with the specified half cycle durations. Detailed, two bed simulation requires the system time step to be less than the individual cyclic intervals and OPS logic is required to properly cycle the air flow paths to the active components. Process cabin air is passed through the assembly via streams 1 and 2 where the water vapor mass transfer rate is calculated as a function of the equilibrium isotherm data (i.e. bed loading and transient packed bed temperature). The removed water vapor leaves via stream 3 during the desorb half-cycle. The DEHUM icon defining the respective streams is shown below.



10.17.1 General Discussion

The analytical approach to modeling the process is based on the determination of the mass transfer rate as a function of the equilibrium isotherm data. An R-C network technique is utilized for both thermal and water vapor density calculations and is discussed in detail in the molecular sieve routine description (section 10.28). Functional schematics of the networks are provided in figure 60. Coding of the two routines MOLSIEV and DEHUM are identical with the following exceptions.

A. The constituents with which the components operate (MOLSIEV – CO_2, DEHUM – H_2O).

B. The MOLSIEV thermal vacuum desorb logic has been updated to include the following:

1. A bed interstitial gas pressure node. DEHUM simply assumes this pressure equal to the bed equilibrium partial pressure at the current bed loading and temperature.

2. Logic to calculate the pressure and inlet. DEHUM currently assumes the vacuum pump inlet conditions are equal to the bed outlet conditions.

3. The MOLSIEV allows the user the option of specifying the performance data for the vacuum pump assembly either by:

   (a) volumetric flow as a function of pressure ratio, pump displacement, clearance factor and polytropic exponent (this is also the current DEHUM baseline), or;

   (b) pressure drop as a function of volumetric flow for a series arrangement of a vacuum pump and compressor in series.

Future releases of DEHUM will incorporate updates similar to those of the MOLSIEV.

10.17.2 Input Parameters

The input parameters for DEHUM are presented in table 39.

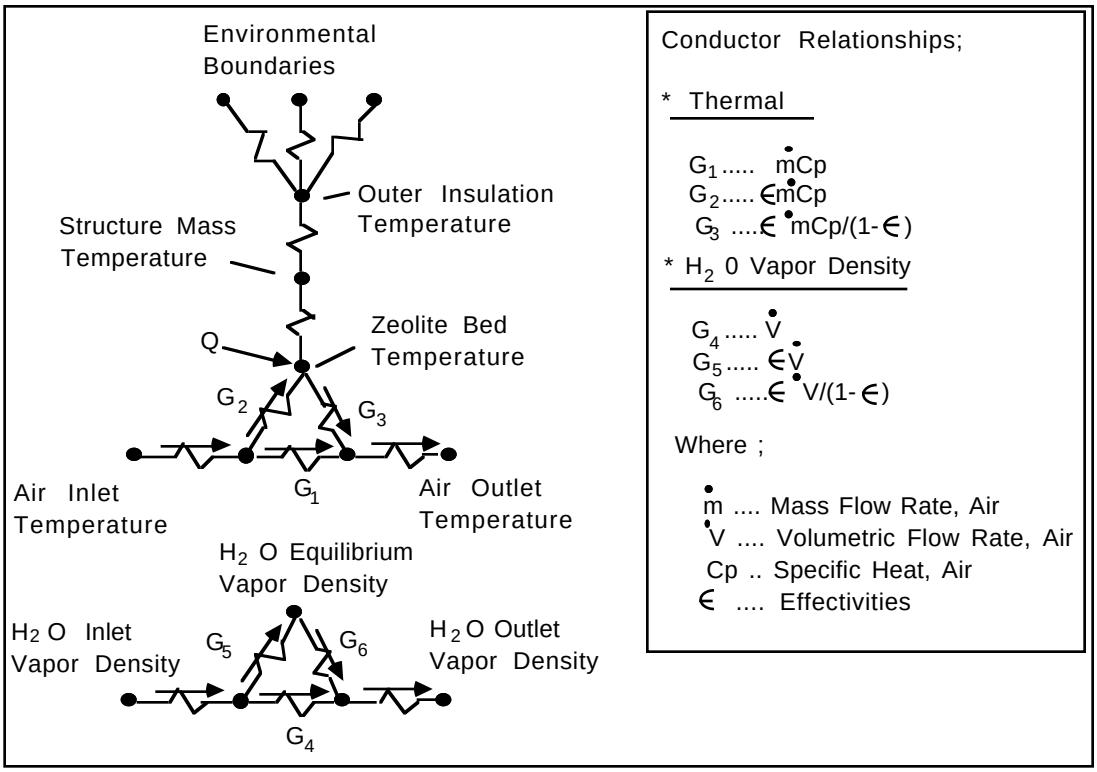


Figure 60. DEHUM thermal/vapor density networks.

Table 39. DEHUM edit screen.

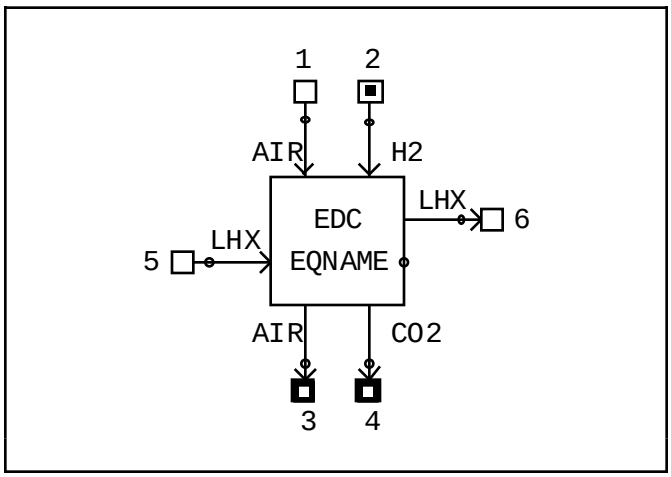
```
CASE NAME: DEFAULT                                    DEHUM EDIT SCREEN                                    LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                      RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                              ** GENERAL INPUT DATA **                                  |     ** FLUID COOLING PASS PARAMETERS **      |
| 38 BED FLAG(0-10X,1-13X,2-SIL)    0.00 | 37 HEATER/VAC FLAG (0-ON,1-OFF)     0.00 | 21 EQUIVALENT LENGTH            (FT)    1.00 |
| 10 ZEOLITE BED WEIGHT    (LBM)   15.88 | 25 INITIAL BED TEMPERATURE   (F)   70.00 | 22 EQUIVALENT DIAMETER          (IN)    6.00 |
| 26 PARTICLE SURFACE AREA (FT2)   23.00 | 36 START DELAY              (HR)    0.00 |        ** THERMAL INTERFACE DATA **         |
| 35 CANISTER WEIGHT       (LBM)   30.00 |                                          | 18 BED HEAT TRANSFR COEF(BTU/FT2/HR)   45.00 |
| 13 ABSORPTION DURATION    (HR)    0.75 |                                          | 16 STRUCTURAL CONDUCTANCE (BTU/HR/F)    2.20 |
| 14 DESORPTION DURATION    (HR)    0.75 |        ** MATERIAL CONSTANTS **          | 15 CONVECTION CONDUCTANCE (BTU/HR/F)   70.00 |
| 20 ABS/DES FLAG(0=ABS,1=DES)      0.00 | 11 HEAT OF ADSORPTN (BTU/LB CO2)  450.00 | 17 ENVIRNMENT RADIATN (BTU/HR R4)  0.100E-09 |
| 24 ACCUMULATOR PRESS    (MMHG)  100.00 | 19 H2O MASS TRNS COEF(LB/FT2/HR)   45.00 | 23 INSULATION CONDUCTANCE (BTU/HR/F)    0.50 |
| 12 BAKEOUT SETPOINT TEMP   (F)  360.00 | 31 POLYTROPIC EXPONENT              1.75 |                                              |
| 28 CONTROLLER GAIN                0.10 |         ** RESOURCE TRACKING **          |    ** SUB-SOLUTION CONTROL PARAMETERS **     |
| 27 MAXIMUM HEATER POWER    (W) 1000.00 | 9 VOLUME                  (FT^3)    2.00 | 34 MAX ITERATIONS                     100.00 |
| 29 PUMP DISPLACMNT   (CID/REV)   80.00 | 8 WEIGHT                   (LBM)   50.00 | 33 DAMPING FACTOR                       0.50 |
| 30 CLEARANCE FACTOR               0.00 | 7 POWER                  (WATTS)  100.00 | 32 TEMPERATURE RELAXATION        (F)    0.10 |
|_____________________________________________________________________________________________________________________________|
|                      ** OUTPUT DATA **                  |     ** BENCHMARK DATA **              MIN         NOM         MAX    |
|          42 MAXIMUM H2O BED LOADING     (%)        0.00 |     AIR IN       (LBM/HR)             0.00        0.00        0.00   |
|          43 NUMBER OF CYCLES          (LBM)        0.00 |     AIR OUT      (LBM/HR)             0.00        0.00        0.00   |
|          44 TOTAL H2O REMOVED         (LBM)        0.00 |     H2O OUT      (LBM/HR)             0.00        0.00        0.00   |
|          45 FINAL H2O BED LOADING     (LBM)        0.00 |                                                                      |
|_____________________________________________________________|_____________________________________________________________________|
```

10.18 EDC—Electrochemical Depolarized CO_2 Concentrator Simulation Description

The EDC routine simulates carbon dioxide (CO_2) concentration from an air stream containing metabolic CO_2. The reaction is summarized stoichiometrically as follows:

$$O_2 + 2CO_2 + 2H_2 \rightarrow 2CO_2 + 2H_2O \ .$$

The CO_2 removal rate is determined by the incoming air flow rate and user input EDC cell removal efficiency. The user may also specify the percent of excess hydrogen in the stream 2 feed and percent of water vapor exiting in stream 4 product. Air leaves the process via stream 3 less the carbon dioxide and water vapor transferred to stream 4. Active component cooling requires the user to specify a steady-state cell operating temperature.



1.18.1 General Discussion

Air enters the system at ambient temperature with low CO_2 partial pressure, and the CO_2 is removed continuously by a liquid phase electrochemical reaction. The CO_2 removal takes place in a module consisting of a series of electrochemical cells. The cells consist of a cathode and anode separated by a membrane containing an aqueous solution of carbonate ions. As the CO_2 laden air comes in contact with the carbonate solution, a liquid phase chemical reaction takes place that provides the following products: (1) concentrated CO_2, (2) electrical energy, and (3) heat. The heat produced by the reaction is removed by continuous flow of cooling fluid through the cells. Waste heat from this process is rejected to the external cooling loop via a liquid/liquid HX. Detailed analysis of the mass and thermal balances are discussed in the following paragraphs.

10.18.2 Mass And Energy Balance

The routine bases CO_2 removal rates on the incoming air flow rate, percent of CO_2 in air, and the process efficiency. For a given inlet air flow rate, the routine calculates an input current value. An energy balance around the EDC module results in the following energy terms: (1) sensible heat of incoming air stream, (2) sensible heat of exiting gases, (3) heat of chemical reaction, and 4) power produced by the chemical reaction. This routine simulates the EDC thermal system by using a delta network to simulate heat transfer within the cells and a four pole network to model the regenerative liquid/liquid HX (fig. 61). The net Q resulting from the summation of energy terms presented above is applied to the delta network configuration. The module cooling required to reject the net heating rate is dependent upon the following user input constants: (1) EDC module cooling effectivity, (2) liquid/liquid HX effectivity, and (3) cooling water flow rate to the cells. The cooling effectivities for this system can be calculated from the following equation:

$$\varepsilon = 1 - e^{-(hA/\dot{m}C_p)} \ ,$$

where,

h = localized heat transfer coefficient (Btu/h/ft^2/°F)

A = surface area (ft^2)

$\dot{m}$ = mass flow rate of cooling fluid (lbm/h)

C_p = specific heat of cooling fluid (Btu/lbm/°F).

For a given set of operating conditions, the program will read the user specified EDC cell operating temperature from the data definition file. The internal EDC subroutine logic is set up to regulate cell temperature by: (1) sensing the incoming coolant flow rate and upstream temperature, (2) calculating the amount of coolant required to keep cell temperatures at the specified level and bypassing the excess, and (3) calculating

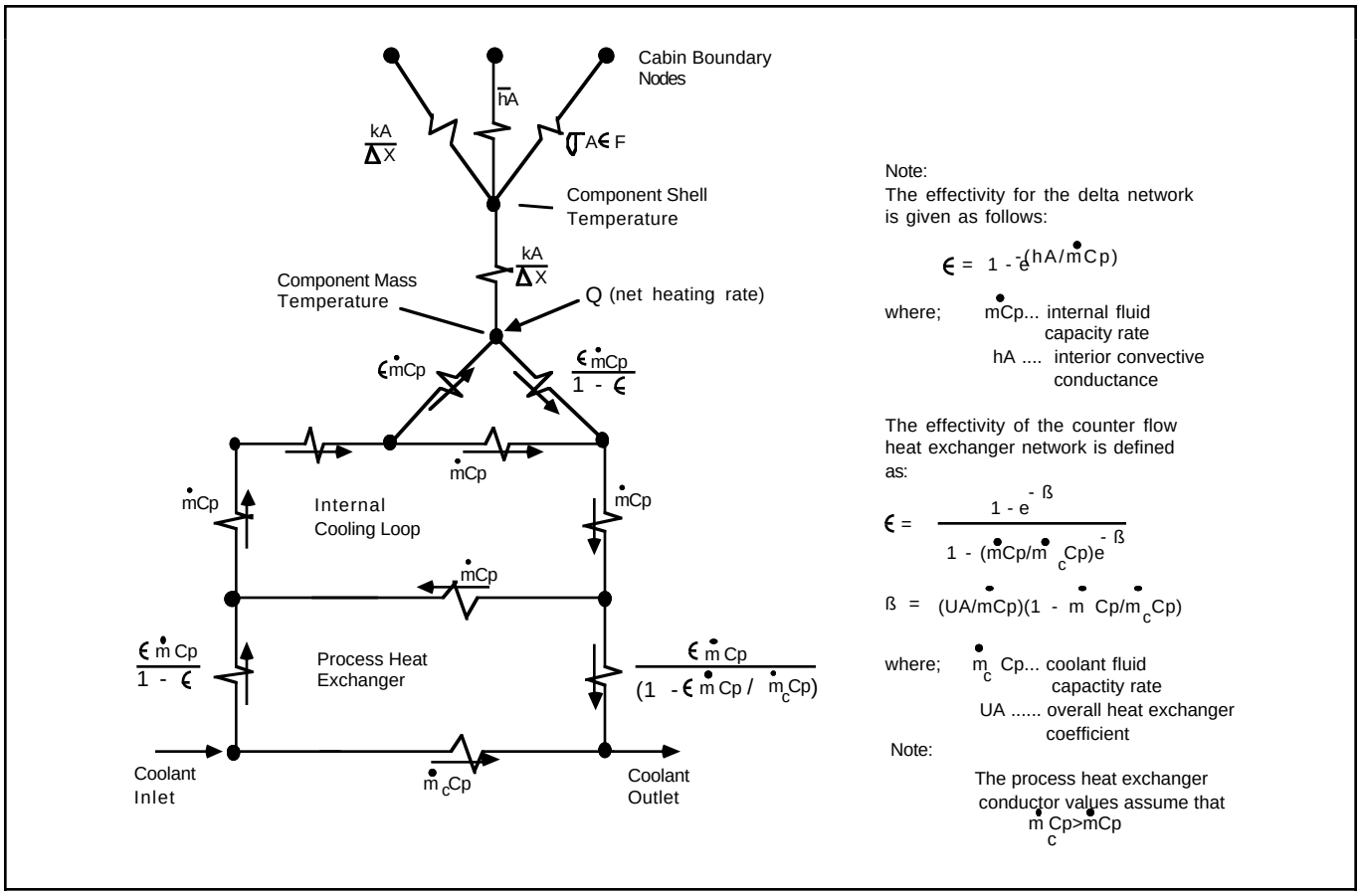


Figure 61. EDC thermal network.

a corresponding mix temperature between coolant fluid exit and coolant fluid bypass before returning the coolant parameters to the external coolant loop. The corresponding cell temperature and coolant flow rate required are written to the output file for each time step. The program assumes the outlet air and concentrated CO_2 streams exit at the prevailing cell temperature. The exiting cooling fluid temperature is calculated as a mass weighted conductor temperature average between the cell steady-state temperature and the cooling entrance temperature.

At steady state, three modes of heat transfer from the cell structure to the cabin environment are provided. Convection, conduction, and radiation conductors are calculated from the user provided constants. After steady-state temperature is converged upon, heat transfer (supplied or extracted) to the cabin is calculated based on these conductor values.

Pressure drop through the EDC module is determined through four user input variables. The outlet pressures for product air and carbon dioxide streams are left to the user for specification. The user must specify a product pressure lower than the total cell pressure. The external liquid coolant stream pressure drop is calculated based on user input equivalent length and diameter values for the EDC module HX. Pressure drop in the liquid stream depends on the Reynolds number and Fanning friction factor that are discussed in section 8.2.4 of this manual.

Operation of the EDC module during steady state results in product stream temperatures near ambient. A user input operating outlet stream temperature is provided so that an error message will be printed out if the current cell temperature exceeds this operating value during program execution.

10.18.3 Input Parameters

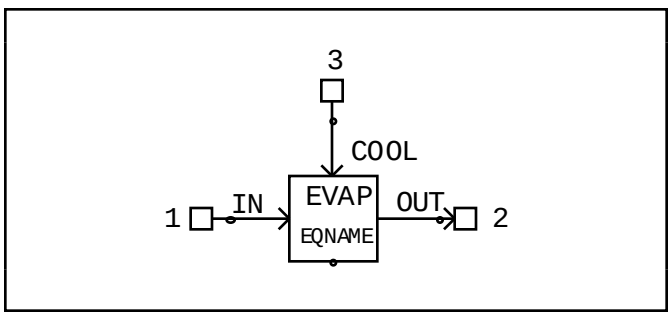
The user input parameters for the EDC routine are shown in table 40.

Table 40. EDC edit screen.

```
CASE NAME: DEFAULT                                    EDC EDIT SCREEN                                          LAST UPDATED: 950321
SUBSYSTEM:                COMPONENT:                                                                          RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                          ** GENERAL INPUT DATA **                                |                                                |
| AIR OUTLET PRESS (PSIA)          50.00 | EDC MODULE COOLING EFF          0.50000 | INITIAL MODULE BYPASS (0-1)           0.60000  |
| CO2 OUTLET PRESS (PSIA)          50.00 | LIQ-LIQ HX EFF                  0.85000 | MAX INTERNAL COOLING FLOW (#/HR)       500.00  |
|                                        |                                         |                                                |
| % MASS H2O IN H2/CO2 PROD        0.000 | VOLUME(FT^3)                       0.00 |                                                |
| % CO2 CONVERSION                34.000 | WEIGHT (LB)                        0.00 |          ** THERMAL INTERFACE DATA **          |
| % EXCESS H2 IN                   0.000 | POWER (WATTS)                      0.00 |  SURFACE AREA (FT^2)                    12.00  |
|                                        |                                         |  CONVECTION COEFF (BTU/HR-FT^2-F)       0.200  |
| VOLTAGE DROP PER CELL (VOLTS)     0.40 |   ** FLUID COOLING PASS PARAMETERS **   |  FORM FACTOR*EMMISSITIVITY              0.800  |
| CELL OPERATING TEMP (F)          80.00 |  EQUIV LNGTH (FT)                 10.00 |  LINEAR CONDUCTANCE (BTU/HR/F)          1.500  |
|                                        |  EQUIV DIA (IN)                    0.50 |                                                |
|_____________________________________|_____________________________________|_____________________________________________|
|                        ** OUTPUT AND BENCHMARK DATA **                                                 MIN        NOM        MAX      |
|  CUM CO2 REMOVED (LBM)          0.00  TRANS CO2 REMOVED (LBM/HR)       0.00  AIR IN     (LB/HR)  0.00E+00   0.00E+00   0.00E+00   |
|  CUM O2 CONSUMED (LBM)          0.00  TRANS O2 CONSUMED (LBM/HR)       0.00  H2 IN      (LB/HR)  0.00E+00   0.00E+00   0.00E+00   |
|  CUM H2 CONSUMED (LBM)          0.00  TRANS H2 CONSUMED (LBM/HR)       0.00  AIR OUT    (LB/HR)  0.00E+00   0.00E+00   0.00E+00   |
|  CUM H2O PRODUCED (LBM)         0.00  TRANS H20 PRODUCED (LBM/HR)      0.00  H2/CO2 OUT (LB/HR)  0.00E+00   0.00E+00   0.00E+00   |
|  TOTAL POWER CONS (WATTS)       0.00  CURRENT MODULE BYPASS            0.00                                                      |
|  TOTAL HEAT GEN (BTU/HR)        0.00  H2O REQD FOR CELLS (LB/HR)       0.00                                                      |
|  ACTIVE CELL AREA (FT^)         0.00  OPERATINTG CELL TEMP (F)         0.00     CURRENT REQD (AMPS)      0.00                    |
|_____________________________________________________________________________________________________________________________|
```

10.19 EVAP—Liquid Flash Evaporator Simulation Description

This routine provides for the steady-state simulation of a liquid flash evaporator configured to utilize the latent heat of vaporization of a working fluid (liquid H_2O) to remove heat from a control fluid. The routine utilizes specified inlet flow rates, specific heats, and temperatures in conjunction with the user specified back pressure (working fluid) to calculate the outlet temperatures of both streams and the state (liquid, saturated vapor/liquid or superheated vapor) and quality, if applicable, of the working side outlet. Streams 1 and 2 are defined as the control fluid inlet and outlet streams, respectively. Stream 3 is the working side liquid feed. The outlet of the working side is assumed to be dumped overboard and is not presently modeled as a CASE/A stream because of two phase flow implications. The EVAP icon defining the respective streams is shown below.



The routine will notify the user if the current conditions result in a working side fluid state either below the triple point or in the compressed liquid region.

10.19.1 General Assumptions

A functional schematic of the thermal network utilized by the EVAP routine is presented in figure 62. A delta network similar to that used for the coldplate component is used to model the control fluid side heat transfer characteristics. The average plate temperature onto which the fluid is sprayed is assumed equal to the average inner housing temperature and the outlet temperature of the working fluid. The effectivity of the control fluid side is a user specified input as is the insulation and environmental conductances.

10.19.2 Mass and Energy Calculations

The method of solution utilized is complicated by the possibility of different working fluid outlet states. An initial assumption of saturated vapor/liquid is made and the saturation temperature corresponding to the user specified back pressure is taken from internal data arrays. Utilizing nodal energy balances the outer housing insulation temperature, the control fluid outlet temperature, and the working fluid outlet enthalpy are calculated;

$$T_s = \frac{G_4T_p+G_5T_{\text{conv}}+G_6T_{\text{cond}}+G_7T_{\text{rad}}}{G_4+G_5+G_6+G_7} \ ,$$

$$T_{co} = \frac{G_1T_{ci}+G_3T_p}{G_1+G_3} \ ,$$

$$h_{wo} = [\dot{m}_w h_{wi}+G_2(T_{ci}-T_p)+G_4(T_s-T_p)]/m_w \ .$$

The respective delta network conductances are calculated by:

$$G_1 = \dot{m}_c c_{pc} \ ,$$

$$G_2 = \varepsilon G_1 \ ,$$

$$G_3 = [\varepsilon/(\varepsilon-1)]\,G_1 \ ,$$

where,

T_s = outer insulation shell temperature, ˚F

T_p = plate/inner housing temperature, ˚F

T_{conv} = mean environ. air temperature, ˚F

T_{cond} = mean environ. structure temperature, ˚F

T_{rad} = mean environ. radiation temperature, ˚F

T_{ci} = control fluid inlet temperature, ˚F

T_{co} = control fluid outlet temperature, ˚F

h_{wi} = working fluid inlet enthalpy, Btu/lbm

h_{wo} = working fluid outlet enthalpy, Btu/lbm

$\dot{m}_w$ = working fluid flowrate, lbm/h

$\dot{m}_c$ = control fluid flowrate, lbm/h

c_{pc} = control fluid constant pressure specific heat , Btu/lbm/˚R

ε = control side effectivity

G_4 = user specified insulation conductance, Btu/h/˚F

G_5 = user specified environ. convection conductance, Btu/h/˚F

G_6 = user specified environ. conduction conductance, Btu/h/˚F

G_7 = linearized radiation conductance, Btu/h/˚F.

Saturated liquid and vapor enthalpies taken from internal arrays are utilized to calculate the outlet quality;

$$x = (h_o - h_{fo})/h_{fgo} \quad ,$$

where,

x = working fluid outlet quality

h_{fo} = Btu/lbm

h_{fgo} = working fluid outlet latent heat of vaporization, Btu/lbm.

If the calculated quality is between $0.0 \le x \le 1.0$, then the assumption of saturation is correct for the given inlet conditions. If the quality is $x < 0.0$, then the outlet state is compressed liquid. The routine will then effectively bypass the EVAP logic ignoring heat loss to the environment and print a message to the screen notifying the user. If the calculated quality is $x > 1.0$, then the outlet state of the working fluid is superheated vapor and the working fluid outlet temperature is calculated iteratively as described in the following section.

The initial guess at the outlet temperature is taken as the saturation temperature. The outlet enthalpy corresponding to the given temperature is taken from an internal data array. The outlet temperature is then calculated from an energy balance around the evaporator plate node:

$$T_p = \frac{m_w(h_{wi} - h_{wo}) + G_2 T_{ci} + G_4 T_s}{G_2 + G_4} \; .$$

The calculated temperature is compared to the guessed temperature with convergence assumed when the difference between the two is below the user specified relaxation. A new guess is made, if required, by the CASE/A utility CONV. The nodal energy balances used in the saturated segment of code are utilized within the superheat segment iterative method to calculate the remaining temperatures of interest. Also the outlet enthalpy is determined from internal arrays during each iteration.

10.19.3 Pressure Drop

The pressure drop through the control fluid side of EVAP is calculated based on user specified equivalent lengths and diameters in the subroutine PIPEDP discussed in section 8.2.5. The pressure drop across the working fluid side is defined by the inlet pressure and the user specified outlet pressure. The appropriate valving is assumed internal to the EVAP to maintain this difference.

10.19.4 Input Parameters

The input parameters for EVAP are presented in table 41.

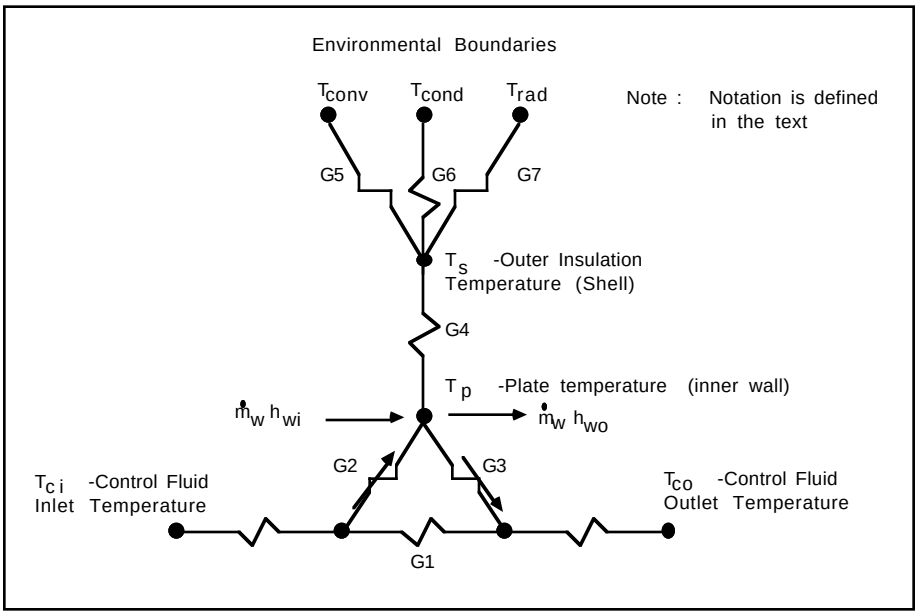


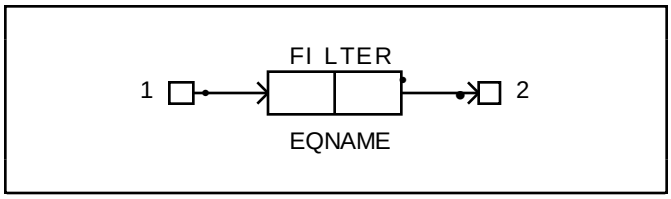
Figure 62. EVAP thermal network.

Table 41. EVAP edit screen.

```
CASE NAME: DEFAULT            EVAP EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:              COMPONENT:                      RECORD NUMBER: 000001
_____________________________________________________________________________
|                              ** GENERAL INPUT DATA **                     |
|10 FEED TYPE (1-H2O,2-NH3)       1.0 |12 INSUL CONDUCTANCE (BTU/HR/F)  10.000 |
|                                     |13 CONV CONDUCTANCE  (BTU/HR/F)  10.000 |
|15 INITIAL OUTLT PRES (PSIA)    2.50 |11 RADIATIVE F-A-E       (FT^2)   5.000 |
|17 EQUIVALENT LENGTH    (FT)   50.00 |14 STRUCT COND KA/X  (BTU/HR-F)  10.000 |
|18 EQUIVALENT DIAMETER  (IN)    0.50 |16 FLUID WALL Q TRANS EFF 0<X<1 0.90000 |
|                                     |          ** RESOURCE TRACKING **       |
|                                     | 7 POWER                (WATTS)    0.00 |
|                                     | 8 WEIGHT                 (LBM)    0.00 |
|                                     | 9 VOLUME                (FT^3)    0.00 |
|_____________________________________|________________________________________|
|                        ** BENCHMARK AND OUTPUT DATA **                      |
|                     MIN            NOM            MAX                      |
|   OUTLET QUALITY   0.00E+00       0.00E+00       0.00E+00                   |
|   OUTLET TEMP (F)  0.00E+00       0.00E+00       0.00E+00                   |
|20 OUTPUT STATE(1-LQD,2-SAT,3-VAP)       0.00                               |
|21 OUTLET QUALITY                        0.00                               |
|22 OUTLET TEMPERATURE            (F)     0.00                               |
|23 AVERAGE HEAT LOSS         (BTU/HR)    0.00                               |
|27 FINAL OUTLET PRESSURE       (PSIA)    0.00                               |
|_____________________________________________________________________________|
```

10.20 FILTER—Filter Simulation Description

The FILTER routine will predict the performance of porous medium filters. The user must determine the medium performance coefficients for a given set of operating conditions and filter configuration. From these data, the routine will predict the performance of the filter subject to the assumptions made in the development of the governing equations. The FILTER component is a "passive" device with respect to mass flow—it only accepts the upstream mass flow or relies on a downstream component to pull flow through it. Therefore, an "active" flow-generating component such as a PUMP must be located somewhere upstream or downstream of the FILTER to drive the flow through it. The icon configuration is shown below.



10.20.1 General Assumptions

The governing equations were obtained or developed from reference 1. It should be noted that the model for the cake resistance is applicable only over a limited range of pressure drops. Also, since the feed concentration affects the formation of the cake, the cake constants should be calculated from experimental test conditions that will approximate those of the simulation.

10.20.2 Mass and Pressure Calculations

The user inputs include 53 medium coefficients, two cake constants, and 13 configuration variables.

The configuration variables consist of the pressure drop at which filter changeout is to occur, the maximum solids holding capacity of the filter housing that will result in filter changeout if exceeded, the filtration area of the medium, and 10 heat transfer parameters.

The medium coefficients consist of the separation efficiency of the clean medium for each constituent, the clean medium flow resistance constant, and three medium constants determined at maximum blinding of the medium. Each input is discussed below.

The separation efficiencies are evaluated with the medium as clean as possible (at low cumulative volume). Blinding of the medium and subsequent increase in separation efficiencies will be taken into account as discussed below. All efficiencies are entered as fractions and are calculated as follows:

$$E_S = 1 - \frac{X_O}{X_i} \ ,$$

where,

E_S = separation efficiency fraction for individual constituent

X_O = constituent concentration in outlet stream, ppm

X_i = constituent concentration in inlet stream, ppm.

The flow resistance of the clean medium is evaluated by the following equation:

$$R = \frac{g_C A \Delta P}{\mu Q} \ ,$$

where,

A = filtration area (ft^2)

ΔP = pressure drop across filter medium (lbf/ft^2)

μ = fluid viscosity (lbm/ft-h)

Q = volumetric flow rate (ft^3/h)

g_C = Newton's constant = 4.1697504×10^8 ft-lbm/(lbf-h^2).

This constant is evaluated at the beginning of a filtration cycle while the medium is still clean.

The term "blinded medium" refers to the medium at the time of the first formation of the

cake for surface filters and at the maximum solids capacity for the depth filter. The three constants are the blinded medium flow resistance (evaluated with the same equation as for the clean medium), the mass of the accumulated solids at the time of blinding, and the efficiency increase factor at the time of blinding. This factor is determined by calculating an average separation efficiency for all constituents for the blinded medium and dividing it by an average separation efficiency for the clean medium. With this factor, the routine increases the separation efficiencies from their clean medium values to the blinded medium values as a linear function of the total accumulated solids. If the medium reaches maximum blinding very rapidly or if there is very little change in the medium constants, the user may enter the constants at blinding and set the efficiency increase factor equal to 1.0.

The two cake constants are the specific cake resistance at unit applied pressure drop and the cake compressibility index obtained experimentally. The empirical relationship used in the routine for the specific cake resistance at any pressure drop is:

$$\alpha = \alpha_o(\Delta P_c)^n ,$$

where,

α = specific cake resistance (ft/lbm)

α_o = specific cake resistance per unit pressure drop (ft ^{3}/lbm-lbf)

n = cake compressibility index ($0 \leqq n \leqq 1.0$)

ΔP_c = pressure drop across the cake (lbf/ft^2).

The values for a_o and n are derived from two experimental data points (two equations and two unknowns) by the following equations:

$$\Delta P = \Delta P_c + \Delta P_m ,$$

where,

ΔP_c = pressure drop across cake

ΔP_m = pressure drop across medium

ΔP = total pressure drop.

The pressure drop across the medium is derived from the following relation:

$$\Delta P_m = (\mu RQ)/(Ag_c) ,$$

where μ, R, Q, A, and g_c have been previously defined. Thus, ΔP_c is given as:

$$\Delta P_c = \Delta P - \Delta P_m .$$

Once ΔP_c is known, an equation containing both α_o and n is needed. This equation depends on the type of filtration carried out in the experiment. α_o and n are derived by setting up two simultaneous equations from two data points using the applicable equation for the type of filtration carried out in the experiment.

The following equation applies to constant rate filtration:

$$\alpha_o(\Delta p_c)^n = \frac{2g_c A^2 \Delta P}{\mu C V^2} t - \frac{RA}{CV} ,$$

where,

t = elapsed time (h)

C = feed concentration (lbm/ft^3)

V = cumulative flow volume at elapsed time (ft^3)

The following equation applies to constant pressure filtration:

$$(\Delta P_c)^{1-n} = \alpha_o(1-n) \frac{\mu C Q^2}{g_c A^2} t .$$

The following equation applies to variable rate-variable pressure filtration:

$$(\Delta P_C)^{1-N} = A_O(1-N) \frac{MCQ}{G_C A^2} V .$$

This case is complex since ΔP_c, ΔP_m, V, Q, and t are all dependent variables. The pump characteristic curve must be known in order to solve for α_o and n. Therefore, it is recommended that experimental data be examined for one of the other two cases.

10.20.3 Thermal Calculations

An R-C network analysis technique is utilized in the thermal calculations and is based on the generic network discussed in section 8.3.3 that allows for three modes of heat transfer with the environment: convection, conduction, and radiation. The total mass of the FILTER column is lumped together to form the mass diffusion node. The network solution calculates the fluid outlet and structural temperatures based on the user specified conductances, thermal capacitance of the mass node, and environment and fluid inlet boundary temperatures. There are four linear conductances the user must provide as inputs along with the emissive area term for radiation and a fluid-to-wall heat transfer effectivity. Also, the user must specify an initial mass temperature, a relaxation criterion for all temperatures, and a maximum iteration count not to be exceeded in the solution routine. The solution routine also allows for heat loads to the mass and external shell nodes. In the FILTER calculations, the shell node heat load is set to zero and the mass node heat load is set to zero.

10.20.4 Data Base Parameters

The FILTER data base parameters are shown in table 42. The FILTER constituent data base parameters are shown in table 43. The default data is set such that there is no heat exchange with the environment and no thermal capacitance (steady-state thermal solution). The constituent constants are set so that all constituents will pass directly through the filter with no separation.

Reference

1. Svarovski, L., editor: “Solid-Liquid Separation.” Butterworths, 1977.

Table 42. FILTER edit screen.

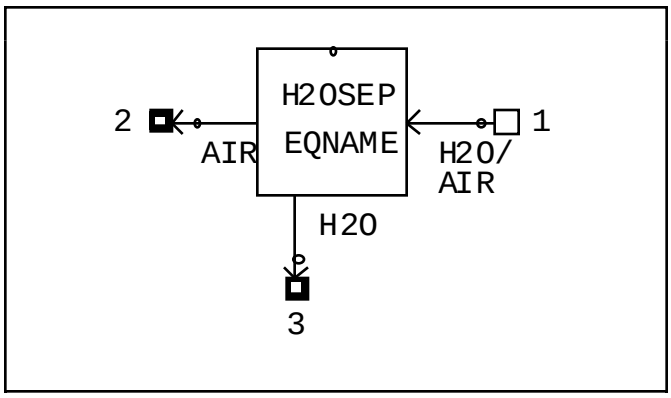
```
CASE NAME: DEFAULT                                  FILTER EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                   RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|          ** GENERAL INPUT DATA **          |    ** THERMAL CHARACTERISTICS DATA **      |     ** MISCELLANEOUS  PARAMETERS **      |
| 10 ACCUM SOLIDS MAX CAP  (LBM)      5.000  | 19 SHELL-ENV CONVCT (BTU/H/F)      0.000   | 29 ** OPEN INPUT LOCATION **      0.000  |
| 11 CHANGEOUT PRESS DROP (PSIA)     50.000  | 20 SHELL-ENV RAD  FAE   (FT2)      0.000   | 30 ** OPEN INPUT LOCATION **      0.000  |
| 12 SPEC CAKE RES CONS (FT/LBM)  0.150E+12  | 21 SHELL-ENV CONDCT (BTU/H/F)      0.000   | 80 ** OPEN INPUT LOCATION **      0.000  |
| 13 CAKE COMPRESS FACT  (0<X<1)  0.2500000  | 22 MASS-SHEL CONDCT (BTU/H/F)      1.000   |                                          |
| 14 CLEAN MED RES CONST  (1/FT)  0.100E+11  | 23 WALL-MASS CONDCT (BTU/H/F)      1.000   |         ** RESOURCE TRACKING **          |
| 15 BLIND MED RES CONST  (1/FT)  0.150E+11  | 24 FLUID-WALL EFFECT  (0<X<1)      0.950   | 7  POWER             (WATTS)     0.0000  |
| 16 CAKE MASS AT BLINDING (LBM)     0.0100  | 25 THERMAL CAPACITNCE (BTU/F)      0.000   | 8  WEIGHT              (LBM)     0.0000  |
| 17 BLINDED SEP EFF FACT (X>=1)     1.0500  | 26 INITIAL MASS TEMPRTURE (F)     75.000   | 9  VOLUME             (FT^3)     0.0000  |
| 18 FILTRATION AREA       (FT2)     1.0000  | 27 THERMAL RELAXATN CRITERIA   0.0000100   |                                          |
| -- CONSTITUENT EMP DATA  (^L  TO OPEN ) _  | 28 THERMAL MAX ITERATN COUNT     100.000   |                                          |
|____________________________________________|____________________________________________|__________________________________________|
|                        ** BENCHMARK DATA **                          |                          ** OUTPUT DATA **                       |
|                          MIN          NOM          MAX               | 83  NUMBER OF UNITS REQUIRED                          0.000        |
|  PRESSURE DROP   (PSID)     0.000        0.000        0.000          | 84  PRESENT PRESSURE DROP            (PSID)           0.000        |
|  PRODUCT  FLOW  (LBM/H)     0.000        0.000        0.000          | 85  PRESENT PRODUCT FLOW RATE       (LBM/HR)          0.000        |
|_____________________________________________________________________| 86  PRESENT MASS  TEMPERATURE           (F)           0.000        |
|  81  PRESENT UNIT'S SERVICE TIME            (HR)      0.000          | 87  PRESENT SHELL TEMPERATURE           (F)           0.000        |
|  82  TOTAL ACCUM SOLIDS FOR ALL UNITS       (LBM)     0.000          | 88  **  OPEN OUTPUT LOCATION  **                          0.000        |
|_____________________________________________________________________|_____________________________________________________________________|
```

Table 43. FILTER constituent edit screen.

```
CASE NAME: DEFAULT       FILTER CONSTITUENT SCREEN        LAST UPDATED: 890215
SUBSYSTEM:               COMPONENT:                      RECORD NUMBER: 000001
_____________________________________________________________________________
|    LABEL    |  SEP EFFCY  |    LABEL    |  SEP EFFCY  |    LABEL    |  SEP EFFCY  |
|             |  FRACTION   |             |  FRACTION   |             |  FRACTION   |
|_____________|_____________|_____________|_____________|_____________|_____________|
| CONSTIT  2  |  0.0000000  | CONSTIT 19  |  0.0000000  | CONSTIT 36  |  0.0000000  |
| CONSTIT  3  |  0.0000000  | CONSTIT 20  |  0.0000000  | CONSTIT 37  |  0.0000000  |
| CONSTIT  4  |  0.0000000  | CONSTIT 21  |  0.0000000  | CONSTIT 38  |  0.0000000  |
| CONSTIT  5  |  0.0000000  | CONSTIT 22  |  0.0000000  | CONSTIT 39  |  0.0000000  |
| CONSTIT  6  |  0.0000000  | CONSTIT 23  |  0.0000000  | CONSTIT 40  |  0.0000000  |
| CONSTIT  7  |  0.0000000  | CONSTIT 24  |  0.0000000  | CONSTIT 41  |  0.0000000  |
| CONSTIT  8  |  0.0000000  | CONSTIT 25  |  0.0000000  | CONSTIT 42  |  0.0000000  |
| CONSTIT  9  |  0.0000000  | CONSTIT 26  |  0.0000000  | CONSTIT 43  |  0.0000000  |
| CONSTIT 10  |  0.0000000  | CONSTIT 27  |  0.0000000  | CONSTIT 44  |  0.0000000  |
| CONSTIT 11  |  0.0000000  | CONSTIT 28  |  0.0000000  | CONSTIT 45  |  0.0000000  |
| CONSTIT 12  |  0.0000000  | CONSTIT 29  |  0.0000000  | CONSTIT 46  |  0.0000000  |
| CONSTIT 13  |  0.0000000  | CONSTIT 30  |  0.0000000  | CONSTIT 47  |  0.0000000  |
| CONSTIT 14  |  0.0000000  | CONSTIT 31  |  0.0000000  | CONSTIT 48  |  0.0000000  |
| CONSTIT 15  |  0.0000000  | CONSTIT 32  |  0.0000000  | CONSTIT 49  |  0.0000000  |
| CONSTIT 16  |  0.0000000  | CONSTIT 33  |  0.0000000  | CONSTIT 50  |  0.0000000  |
| CONSTIT 17  |  0.0000000  | CONSTIT 34  |  0.0000000  |             |             |
| CONSTIT 18  |  0.0000000  | CONSTIT 35  |  0.0000000  |             |             |
|_____________|_____________|_____________|_____________|_____________|_____________|
```

10.21 H2OSEP—Water Separator Simulation Description

The water separator routine is a complementary component simulation to the Condensing Heat Exchanger. This routine separates the liquid water from the two phase stream from the CHX “slurper” and can be used with the WASH components to separate the air/water mixture streams. The air leaves stream 2 saturated, and all power required by the water separator is dissipated in this stream. The pressure for both outlet streams are set by the user. All contaminants are assumed to exit with the water outlet stream (stream 3). This routine is able to handle the condition where no water is contained in the inlet stream by just passing the air completely through the separator.



10.21.1 General Description

The H2OSEP routine determines the quantity of water required to saturate the air at the outlet temperature and sends the remainder to the water outlet stream. The quality of the inlet stream is determined by the properties (PROPS) routine. If the quality is 0.0 then there is no liquid present and the entire inlet flow is directed to the air outlet stream. If the quality is 1.0, that is 100 percent liquid, the air is still saturated and the remaining is sent out the water outlet. The thermal behavior of the water separator is controlled by the user. The user is required to input a power level that the water separator requires. This power will be dissipated to the air stream leaving the separator. The water stream is assumed to be at the same temperature as that of the inlet.

10.21.2 Pressure Calculations

The operating pressures for this unit are specified by the user. The inlet pressure is determined by the upstream component.

10.21.3 Input Parameters

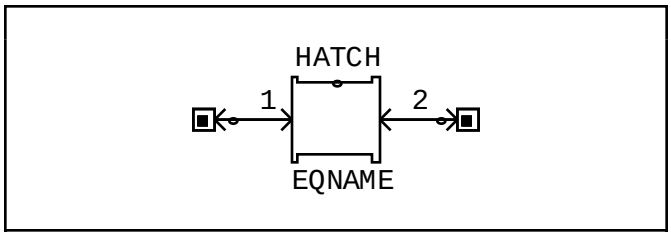
The input parameters are relatively straight forward as shown in table 44. The default values are based on *Spacelab* data.

Table 44. H2OSEP edit screen.

```
CASE NAME: DEFAULT           H2OSEP EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                    COMPONENT:                  RECORD NUMBER: 000001
_____________________________________________________________________________
|             ** GENERAL INPUT DATA **              |   ** RESOURCE TRACKING **   |
| 10 HEAT DISSIPATED TO CABIN        (W)   43.00    |                             |
| 11 OUTLET AIR PRESSURE          (PSIA)   14.90    | 7 POWER   (WATTS)   43.00   |
| 12 OUTLET WATER PRESSURE        (PSIA)   37.00    | 8 WEIGHT    (LBM)    5.60   |
|                                                   | 9 VOLUME   (FT^3)   0.000   |
|___________________________________________________|_____________________________|
|                              ** BENCHMARK DATA **                                |
|                           MIN        NOM        MAX                             |
| AIR/H2O IN    (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                          |
| AIR OUT       (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                          |
| H2O OUT       (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                          |
|_____________________________________________________________________________|
```

10.22 HATCH—HATCH Simulation Description

The HATCH routine simulates the pressure equalization and air exchange functions of a hatch connecting two cabins. It is designed to adjust the flow of air between the cabins so that their pressures are in equilibrium to within the specified tolerance in the HATCH data file. The flow of air can be in either direction, depending on the cabin pressures. The status of the hatch is set by the user, via the input status flag, as either open or closed. If the hatch is closed, no air exchange occurs. The icon configuration is shown below.



10.22.1 General Assumptions

The following assumptions were made in the development of the HATCH routine:

A. The mass of the air in the hatch passage is negligible.

B. No heat transfer with the hatch passage structural walls occurs.

C. No air leakage from the hatch to space occurs.

10.22.2 Mass and Pressure Calculations

The air flow direction and magnitude are calculated based on simple proportional control logic using the pressure differential as the controlling variable. The airflow is adjusted in an iterative process until convergence is obtained according to the following equation:

$$\dot{m}'_{2-1} = \dot{m}_{2-1} + \gamma(P_2 - P_1) ,$$

where,

$\dot{m}'_{2-1}$ = new calculated air flow rate from cabin No. 2 to cabin No. 1.

$\dot{m}_{2-1}$ = last iteration air flow rate from cabin No. 2 to cabin No. 1.

γ = differential pressure gain factor

P_1 = cabin No. 1 air pressure

P_2 = cabin No. 2 air pressure.

A few iterations are usually required to obtain convergence since this type of solution is a variation of the back substitution solution technique. The HATCH merely passes the air composition of the CABIN from which the air is being removed to the second CABIN.

10.22.3 Thermal Calculations

There are no heat transfer calculations carried out in the HATCH routine. The outgoing air temperature is set equal to the incoming air temperature.

10.22.4 Data Base Parameters

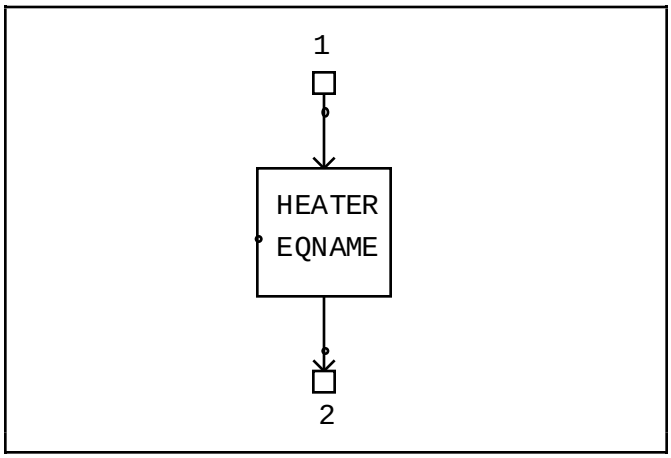
The HATCH data base parameters are shown in table 45. The default setting of the status flag is set to zero, which corresponds to the open state.

Table 45. HATCH edit screen.

```
CASE NAME: DEFAULT            HATCH EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                   COMPONENT:                   RECORD NUMBER: 000001
|                                ** INPUT DATA **                                |
|10 STATUS FLAG       (0=OPEN, 1=CLOSED)     0.0   7 POWER   (WATTS)    0.00   |
|11 PATHWAY DIAMETER                (IN)   55.00   8 WEIGHT    (LBM)    0.00   |
|12 PATHWAY LENGTH                  (IN)   12.00   9 VOLUME   (FT^3)    0.00   |
|13 FLOW GAIN FACTOR       (.01 <= 0.99)  0.1000  14 PDEL RELX(PSIA)  0.0000   |
|                       MIN        NOM        MAX      ** BENCHMARK DATA **     |
| FLOW1-2(LBM/HR)  0.0000E+00  0.0000E+00  0.0000E+00                           |
| FLOW2-1(LBM/HR)  0.0000E+00  0.0000E+00  0.0000E+00                           |
| VELOCTY(FT/SEC)  0.0000E+00  0.0000E+00  0.0000E+00                           |
|15 CURRENT TEMP              (F)  0.0000E+00          ** OUTPUT DATA **        |
|16 CURRENT PRESSURE       (PSIA)  0.0000E+00                                   |
|17 CURRENT FLOW VELOCTY (FT/SEC)  0.0000E+00                                   |
|18 CURRENT FLOW1-2      (LBM/HR)  0.0000E+00                                   |
|19 CURRENT FLOW2-1      (LBM/HR)  0.0000E+00                                   |
```

10.23 HEATER—HEATER Simulation Description

This routine simulates the performance of a flow-through heater (HEATER). The principal user input is the desired outlet fluid temperature. The HEATER does not generate flow, therefore an active component must be upstream of the inlet. The HEATER data base parameters are shown in table 46.



Discussion

The user selects the desired outlet fluid temperature and the routine utilizes conventional energy equations to calculate the heat rate required (Btu/h) to achieve the selected outlet temperature. If the inlet fluid temperature is above the desired outlet temperature, then no power consumption occurs and the outlet temperature equals the inlet temperature. Also, a message appears to the screen notifying the user of the condition. Nominal power consumption is calculated by dividing the total heat required (Btu) by the cumulative time that heater power was required.

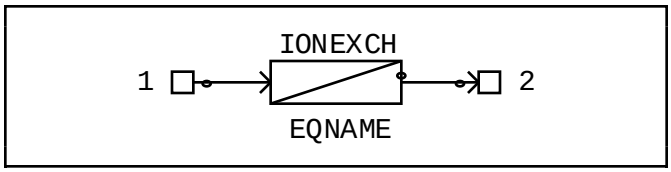
Pressure drop through the HEATER is found by conventional pressure drop calculation methods utilizing a user-supplied equivalent length and equivalent diameter. See section 8.2 for a detailed explanation of the pressure drop calculations.

Table 46. HEATER edit screen.

```
CASE NAME: DEFAULT            HEATER EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                    COMPONENT:                   RECORD NUMBER: 000001
_____________________________________________________________________________________
| ** INPUT DATA                                                                     |
| 10 DESIRED FLUID OUTLET TEMP (F)   70.00 |       * RESOURCE TRACKING *            |
|                                          | 7 POWER           (WATTS)    0.00 |
|        * PRESSURE DROP PARAMETERS *      | 8 WEIGHT            (LBM)    0.00 |
| 11 EQUIVALENT LENGTH        (FT)   10.00 | 9 VOLUME           (FT^3)    0.00 |
| 12 EQUIVALENT DIAMETER      (IN)    0.75 |                                      |
|__________________________________________|______________________________________|
| ** BENCHMARK DATA                                  | ** OUTPUT DATA                |
|                       MIN        NOM        MAX    |                               |
| INLET TEMP     (F)  0.00E+00   0.00E+00   0.00E+00|13 PWR (BTU/HR) XX   0.00 |
| INLET FLOW (LB/HR)  0.00E+00   0.00E+00   0.00E+00|14 OPER TIME (HR)    0.00 |
| HEAT REQD (BTU/HR)  0.00E+00   0.00E+00   0.00E+00|                               |
|____________________________________________________|_______________________________|
```

10.24 IONEXCH—Ion Exchange Resin Simulation Description

The IONEXCH routine simulates the performance of a cylindrical ion exchange unit. The user provides the performance characteristics for a given resin and ion exchange unit size. The routine will then predict the performance and operational life of the unit. The ion exchange process is modeled as a special case of the adsorption process. The routine logic is identical to that of the ADSORPTN routine. The reader should refer to section 10.1 for the simulation description.



Data Base Parameters

The IONEXCH main edit screen is shown in table 47. The IONEXCH constituent data base parameters are shown in table 48. The default data is set such that there is no heat exchange with the environment and no thermal capacitance (steady-state thermal solution). The constituent constants are set so that all constituents will pass directly through the column with no adsorption.

Table 47. IONEXCH edit screen.

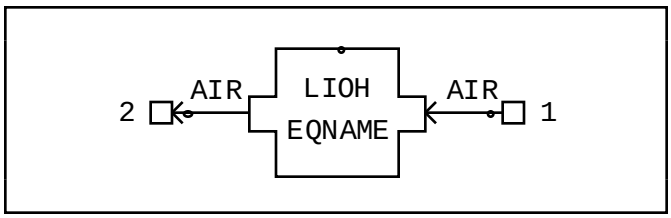
```
CASE NAME: DEFAULT                                   IONEXCH EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|            ** GENERAL INPUT DATA **          |    ** THERMAL CHARACTERISTICS DATA **     |     ** MISCELLANEOUS  PARAMETERS **     |
| 10 IONEXCH COLUMN DIAMETER (FT)     0.250    | 15 SHELL-ENV CONVCT (BTU/HR/F)     0.000 | 25  ** OPEN INPUT LOCATION **     0.000 |
| 11 IONEXCH COLUMN LENGTH   (FT)     1.000    | 16 SHELL-ENV RAD F*A*E   (FT2)     0.000 | 26  ** OPEN INPUT LOCATION **     0.000 |
| 12 COLUMN FLOW RESIST   (1/FT2) 0.100E+11    | 17 SHELL-ENV CONDCT (BTU/HR/F)     0.000 | 174 ** OPEN INPUT LOCATION **     0.000 |
| 13 TOTAL CAPACITY    (LBM/FT3)    22.640    | 18 MASS-SHEL CONDCT (BTU/HR/F)     1.000 |                                         |
| 14 ADSORP CONSTANT (FT3/LBM/HR)     1.620    | 19 WALL-MASS CONDCT (BTU/HR/F)     1.000 |          ** RESOURCE TRACKING **        |
|                                              | 20 FLUID-WALL EFFECT    (0<X<1)    0.950 | 7  POWER              (WATTS)    0.0000 |
|                                              | 21 THERMAL CAPACITANCE (BTU/F)     0.000 | 8  WEIGHT               (LBM)    0.0000 |
|                                              | 22 INITIAL MASS TEMP       (F)    75.000 | 9  VOLUME              (FT^3)    0.0000 |
|                                              | 23 THERMAL RELAXATN CRITERIA   0.0010000 |                                         |
| -- CONSTITUENT EMP DATA  (^L  TO OPEN ) _    | 24 THERMAL MAX ITERATN COUNT     100.000 |                                         |
|_____________________________________________|_________________________________________|_________________________________________|
|                      ** BENCHMARK DATA **                   |                          ** OUTPUT DATA **                          |
|                          MIN          NOM          MAX      |  177  NUMBER OF UNITS REQUIRED                          0.000        |
|   PRESSURE DROP    (PSID)     0.000        0.000        0.000  |  178  PRESENT PRESSURE DROP              (PSID)     0.000        |
|   PRODUCT  FLOW  (LBM/HR)     0.000        0.000        0.000  |  179  PRESENT PRODUCT FLOW RATE        (LBM/HR)     0.000        |
|_____________________________________________________________|  180  PRESENT MASS TEMPERATURE              (F)     0.000        |
|  175  PRESENT UNIT'S SERVICE TIME              (HR)     0.000  |  181  PRESENT SHELL TEMPERATURE             (F)     0.000        |
|  176  TOTAL ACCUM SOLIDS FOR ALL UNITS        (LBM)     0.000  |  182  **  OPEN OUTPUT LOCATION  **                          0.000        |
|_____________________________________________________________|_____________________________________________________________________|
```

Table 48. IONEXCH constituent edit screen.

```
CASE NAME:  DEFAULT                                  IONEXCH CONSTITUENT SCREEN                           LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|   LABEL    | EMPIRICL | EMPIRICL | BRKTHRU |   LABEL    | EMPIRICL | EMPIRICL | BRKTHRU |   LABEL    | EMPIRICL | EMPIRICL | BRKTHRU |
|            | CNSTNT A | CNSTNT B |  (PPM)  |            | CNSTNT A | CNSTNT B |  (PPM)  |            | CNSTNT A | CNSTNT B |  (PPM)  |
|____________|__________|__________|_________|____________|__________|__________|_________|____________|__________|__________|_________|
|CONSTIT 2   0.000E+00 0.000E+00 2000000.0 CONSTIT 19   0.000E+00 0.000E+00 2000000.0 CONSTIT 36   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 3   0.000E+00 0.000E+00 2000000.0 CONSTIT 20   0.000E+00 0.000E+00 2000000.0 CONSTIT 37   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 4   0.000E+00 0.000E+00 2000000.0 CONSTIT 21   0.000E+00 0.000E+00 2000000.0 CONSTIT 38   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 5   0.000E+00 0.000E+00 2000000.0 CONSTIT 22   0.000E+00 0.000E+00 2000000.0 CONSTIT 39   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 6   0.000E+00 0.000E+00 2000000.0 CONSTIT 23   0.000E+00 0.000E+00 2000000.0 CONSTIT 40   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 7   0.000E+00 0.000E+00 2000000.0 CONSTIT 24   0.000E+00 0.000E+00 2000000.0 CONSTIT 41   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 8   0.000E+00 0.000E+00 2000000.0 CONSTIT 25   0.000E+00 0.000E+00 2000000.0 CONSTIT 42   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 9   0.000E+00 0.000E+00 2000000.0 CONSTIT 26   0.000E+00 0.000E+00 2000000.0 CONSTIT 43   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 10  0.000E+00 0.000E+00 2000000.0 CONSTIT 27   0.000E+00 0.000E+00 2000000.0 CONSTIT 44   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 11  0.000E+00 0.000E+00 2000000.0 CONSTIT 28   0.000E+00 0.000E+00 2000000.0 CONSTIT 45   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 12  0.000E+00 0.000E+00 2000000.0 CONSTIT 29   0.000E+00 0.000E+00 2000000.0 CONSTIT 46   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 13  0.000E+00 0.000E+00 2000000.0 CONSTIT 30   0.000E+00 0.000E+00 2000000.0 CONSTIT 47   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 14  0.000E+00 0.000E+00 2000000.0 CONSTIT 31   0.000E+00 0.000E+00 2000000.0 CONSTIT 48   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 15  0.000E+00 0.000E+00 2000000.0 CONSTIT 32   0.000E+00 0.000E+00 2000000.0 CONSTIT 49   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 16  0.000E+00 0.000E+00 2000000.0 CONSTIT 33   0.000E+00 0.000E+00 2000000.0 CONSTIT 50   0.000E+00 0.000E+00  2000000.0 |
|CONSTIT 17  0.000E+00 0.000E+00 2000000.0 CONSTIT 34   0.000E+00 0.000E+00 2000000.0                                             |
|CONSTIT 18  0.000E+00 0.000E+00 2000000.0 CONSTIT 35   0.000E+00 0.000E+00 2000000.0                                             |
|____________________________________________________________________________________________________________________________|
```

10.25 LIOH—Lithium Hydroxide Simulation Description

The LIOH routine simulates the carbon dioxide removal characteristic of a single LiOH cartridge. Process air is passed through the cartridge via streams 1 and 2. The carbon dioxide removal rate is calculated using the packed bed mass transfer equations applied to an ash diffusion controlled process. Water vapor generated by the exothermic reaction is assumed to be in gaseous equilibrium with the air and remaining carbon dioxide in the outlet stream. A user input internal time step is implemented to insure numerical integration stability of the mass transfer equations. The cartridge is considered to be adiabatic and the heat of reaction is removed by the sensible enthalpy transfer of the outlet air stream.



10.25.1 General Description

The absorption process of removing CO_2 from incoming air involves the conversion of LiOH to Li_2CO_3 and water vapor. The actual conversion occurs through a series of intermediate reactions, however the process can be simulated with satisfactory results by considering the overall net reaction:

$$CO_2 + 2LiOH \rightarrow Li_2CO_3 + H_2O \ .$$

This reaction is exothermic and is classified as a noncatalytic fluid-solid type conversion process. A simple idealized scheme was chosen for modeling the process. The method considers the reaction to occur first at the outer skin of the solid LiOH particle, and then moves into the solid leaving behind completely converted material. This modeling scheme is called the unreacted core method. Some of the basic assumptions necessary to apply this method include the following:

A. The composition of the reactive bed is assumed to be composed of spherical particles that consist of LiOH and Li_2CO_3.

B. The Li_2CO_3 forms an inert ash layer surrounding the unreacted core of LiOH.

C. Overall packed bed particle diameters are constant; therefore, the ash layer depth is calculated as a simple function of the quantity of LiOH reacted at any given point in time.

D. Overall reaction rates are controlled by the mass transfer rate of CO_2 from the packed bed free stream through the inert Li_2CO_3 ash layer to the active LiOH core.

E. Chemical kinetic reaction rates at the active LiOH core are assumed to be large when compared to the ash and film diffusion controls.

The following sections provide a detailed explanation of the mass and energy balance relationships, and numerical solution method utilized for simulation of this process.

10.25.2 Mass Transfer Relationships

Overall CO_2 mass transfer rates for a chemical conversion process of this type may be determined using familiar R-C network techniques. Convective free stream mass transfer mechanisms are simulated using a “one-way” fluid-to-wall delta network arrangement. Molecular diffusion rates through the ash layer are calculated via a standard diffusion network. A functional schematic of the composite mass transfer network (R-C analogous) is shown in figure 63.

The delta segment provides for the computation of the CO_2 mass transfer rates that occur between the convective air free stream and the LiOH particle surface. An effectivity factor is associated with this process and is defined as follows:

$$\varepsilon = 1 - e^{-[kA/\dot{v}]} \ , \qquad (1)$$

where,

k = packed bed convective mass transfer coefficient, ft/h

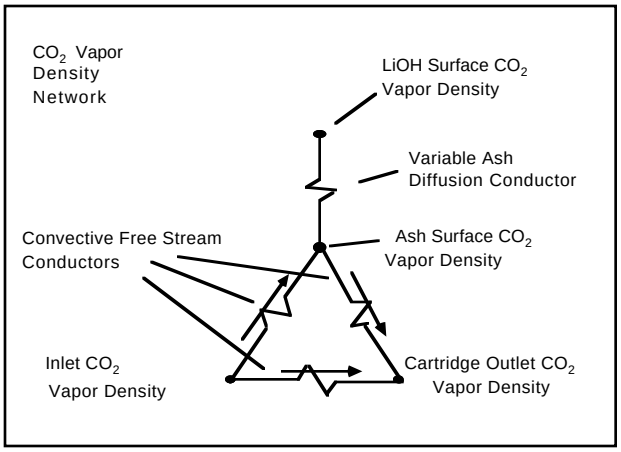


Figure 63. LIOH vapor density network.

A = total particle surface area, ft^2

$\dot{v}$ = volumetric flow of air, ft^3/h.

The overall convective mass transfer coefficient in the above equation is provided as a user specified input constant. The coefficient is, however, a function of the local fluid transport properties (Reynolds, Sherwood, and Schmidt numbers). Its value may be dynamically computed in supplemental OPS logic if desired. The mass transfer network equations for the delta segment are given as follows:

$$0 = \varepsilon\dot{v}(\rho_i - \rho_{ps}) + G_A(\rho_1 - \rho_{ps}), \qquad (2)$$

$$0 = \beta\dot{v}(\rho_{ps} - \rho_o) = \dot{v}(\rho_i - \rho_o), \qquad (3)$$

$$\beta = \left(\frac{\varepsilon}{1-\varepsilon}\right),$$

where,

ρ_i = cartridge inlet CO_2 vapor density (ideal gas calculation based on CO_2 partial pressure), lbm/ft^3

ρ_{out} = cartridge outlet CO_2 vapor density

ρ_{ps} = CO_2 vapor density at particle surface

ρ_1 = CO_2 vapor density at active LiOH core (this value is assumed to be null, i.e., $p_1 = 0$)

G_A = Li_2CO_3 ash conductor, Btu/h °F.

Molecular diffusion rates (the overall controlling factor) through the Li_2CO_3 ash layer are computed as time-dependent quantities via the implementation of a variable "ash diffusion" conductance. Since the bed is assumed to be composed of spherical particles, the conductor developed for use is expressed in spherical coordinates. It is given as:

$$G_A = \left[\frac{4\pi\alpha D_{AB}}{\frac{1}{R_1} - \frac{1}{R_0}}\right], \qquad (4)$$

where,

D_{AB} = effective binary mass molecular diffusivity (CO_2 air), ft^2/h

R_1 = active LiOH core radius (a vanishing quantity), ft

R_o = outer particle radius (a constant quantity), ft

α = porosity attenuation factor.

Derivation and justification for use of the porosity factor may be obtained in reference 1.

The active LiOH core radius (R_1) is considered to be a function of the extent of reaction. That is, it is dependent on the total amount of CO_2 absorbed per cartridge. The differential equation that describes the CO_2 removal rate is expressed as follows:

$$\frac{dm_c}{dt} = \dot{v}(\rho_1 - \rho_o), \qquad (5)$$

where;

m_c = mass of CO_2 converted by reaction, lbm.

The ash layer depth is now calculable as a function of reaction stoichiometry (equation (5)) and spherical particle geometric considerations.

The basic equation utilized in the model for the determination of R_1 is given as:

$$R_1 = R_o\left[\frac{M_B - M(t)}{M_B}\right]^{1/3}, \qquad (6)$$

where,

M_B = LiOH bed weight

$M(t)$ = $M_c/0.92$, amount of LiOH converted in bed, lbm.

As a consequence of the fast chemical kinetic reaction rate, the CO_2 vapor density at the active LiOH core (figure 63) is assumed to be zero. This allows for the calculations of the ash surface and cartridge outlet CO_2 vapor densities in terms of the concentration gradients caused by the ash diffusion controls.

10.25.3 Energy Balance

A parallel R-C thermal network is developed for simulation of the heat transfer characteristics of the packed bed while undergoing a chemical reaction. The schematic of the conductor network is similar to the mass transfer network shown in figure 65. The nodal scheme is identical to that of the mass transfer network and the various segments perform analogous functions. The convective delta segment simulates the heat transfer rates between the particle and the air free stream. The elemental form of equation (2) is preserved, but the constituents of the dimensionless argument has changed from:

$$\left[\frac{kA}{\dot{v}}\right] \rightarrow \left[\frac{\bar{h}A}{\dot{m}Cp}\right],$$

where,

h = overall packed bed convective heat transfer coefficient, Btu/h ft$_2$ °F

$\dot{m}Cp$ = air capacity rate, Btu/h °F.

The ash conductor also conserves its basic form. However, the binary mass molecular diffusivity (D_{AB}) is now replaced with the thermal conductivity of the ash layer.

All nodes in the thermal network are arithmetic and boundary types and cartridge walls are assumed to be adiabatic. The network nodal equations to be solved simultaneously with the mass transfer equations are listed as follows:

$$0 = \varepsilon\dot{m}Cp(T_i - T_{ps}) + G_a(T_1 - T_{ps}), \qquad (7)$$

$$0 = G_A(T_{ps} - T_1) + Q_R, \qquad (8)$$

$$0 = \beta\dot{m}Cp(T_{ps} - T_o) + \dot{m}C_P(T_i - T_o), \qquad (9)$$

where,

T_i = cartridge inlet temperature, °F

T_o = cartridge outlet temperature, °F

T_{ps} = particle surface temperature, °F

T_1 = active LiOH core temperature, °F

$$\beta = \left[\frac{\varepsilon}{1-\varepsilon}\right]$$

Q_R = heat of reaction as determined by CO_2 removal rate.

10.25.4 Numerical Solution

A Gauss-Siedel method of solution is performed on the algebraic equations with the implicit dependent variables temperature and vapor density. The diffusional ash conductors are calculated as functions of the average ash depth across the time step. This requires a predictor-corrector (PC) method of numerical integration to be used on the bed loading differential equation (equation (5)). The explicit PC logic is simultaneously solved with the iterative algebraic equations in the same manner as that used for the SAWD routine.

10.25.5 Input Parameters

The input parameters are shown in table 49. The primary items of interest to the user are the bed mass, equivalent length, and diameter. The cartridge change out flag is used to simulate cartridge change out and are controlled by the user from the OPS logic block. Failure to change out cartridges will result in a loss of CO_2 control in the cabins. The remaining parameters are

dependent on the particle geometry and size. The default values are based on Spacelab analysis from actual flight data.

References

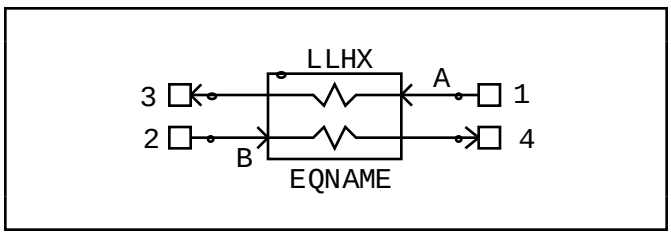
1. Clayton, J.L.: "Analytical Assessment of Lithium Hydroxide (LiOH) Cartridge Performance." McDonnell Douglas TM SEAD-84062, July 1984.

Table 49. LIOH edit screen.

```
CASE NAME: DEFAULT             LIOH EDIT SCREEN          LAST UPDATED: 890215
SUBSYSTEM:              COMPONENT:                      RECORD NUMBER: 000001
_____________________________________________________________________________
|                            ** GENERAL INPUT DATA **                        |
|11 LIOH BED MASS         (LBM)    5.00 |15 VAPOR RELAX COEFFICIENT   0.10E-05 |
|                                       |14 THERMAL RELAX COEFFICIENT 0.10000  |
|12 MASS TRANS COEFF  (FT^2/HR)  380.00 |17 VAPOR DAMPING FACTOR        0.000  |
|13 HEAT TRANS CO (B/HR/FT^2/F)   45.00 |16 THERMAL DAMPING FACTOR      0.100  |
|19 HEAT OF REACTION  (BTU/LBM)  874.00 |18 INTERNAL TIME STEP  (HRS)    0.01  |
|                                       |22 MAXIMUM ITERATIONS          300.0  |
|20 POROSITY ATTENUATION FACTOR 0.00009 |       ** RESOURCE TRACKING **        |
|21 CARTRIDGE CHANGEOUT FLAG        0.0 | 7 POWER             (WATTS)    0.00  |
|24 CARTRIDGE EQUIV LENGTH (FT)    1.00 | 8 WEIGHT              (LBM)    0.00  |
|23 CARTRIDGE EQUIV DIA    (IN)    6.00 | 9 VOLUME             (FT^3)    0.00  |
|_____________________________________|_____________________________________|
|                            ** BENCHMARK DATA **                            |
|                      MIN          NOM          MAX                         |
| CO2 REMOVAL RATE   0.00E+00     0.00E+00     0.00E+00(LBM CO2/HR)           |
| CO2 BED LOADING    0.00E+00     0.00E+00     0.00E+00(LBM CO2 )             |
|                             ** OUTPUT DATA **                               |
|26 CARTRIDGE INDEX              0.0 29 CURRENT CO2 ABSORBED (LBM)    0.00   |
|28 TOTAL H20 PRODUCED (LBM)    0.00 27 TOTAL CO2 ABSORBED   (LBM)    0.00   |
|_____________________________________________________________________________|
```

10.26 LLHX—Liquid/Liquid Heater Exchanger Simulation Description

This routine simulates the thermal control function of a liquid/liquid HX (LLHX). HX analysis is based on the effectiveness-NTU method. HX effectiveness is calculated by the routine utilizing a user-specified overall heat conductance (UA) with cross-flow (both fluids unmixed), counterflow, or parallel-flow equations depending on the desired application. The LLHX does not exchange heat with its environment. One side of the LLHX is arbitrarily designated as side "A" and the other side as side "B" to allow proper pressure drop data input. The LLHX does not generate flow; therefore, an active component must be upstream of each inlet. Pressure drop through the LLHX is determined by user-supplied equivalent line lengths and equivalent line diameters. The LLHX data base parameters are shown in table 50.



Discussion

The LLHX routine is based on the effectiveness-NTU analysis approach of HX performance. This method uses specified inlet flow rates, specific heats, and temperatures, in conjunction with a user-specified overall heat conductance (UA). It is assumed that there is negligible heat transfer between the exchanger and its surroundings, as well as negligible potential and kinetic energy changes, which allow use of the appropriate energy balance equations. The resultant outlet temperatures and heat transfer characteristics are determined as functions of the above data and an internally calculated HX effectiveness. Three distinct types of HX simulation are currently available; parallel-flow, counter-flow, and cross-flow (both fluids unmixed). The HX effectiveness relationships for these geometrically dissimilar flow arrangements are given as follows:

$$\varepsilon = \frac{1-\exp[-\mathrm{NTU}(1+C_r)]}{1+C_r}\,, \qquad \text{(parallel-flow)}$$

$$\varepsilon = \frac{1-\exp[-\mathrm{NTU}(1-C_r)]}{1-C_r\,\exp[-\mathrm{NTU}(1-C_r)]}\,, \qquad \text{(counter-flow)}$$

$$\varepsilon \approx 1-\exp\{\frac{1}{C_r}\mathrm{NTU}^{0.22}[\exp(-C_r\mathrm{NTU}^{0.78})-1]\}\,, \qquad \text{(cross-flow)}$$

where,

$\mathrm{NTU} = \mathrm{UA}/C_{\min}$

$C_r = C_{\min}/C_{\max}$

$C_{\min} = \dot{m}C_p$, minimum fluid heat capacity rate, Btu/h/°F

$C_{\max} = \dot{m}C_p$, maximum fluid heat capacity rate, Btu/h/°F

UA = overall heat conductance, Btu/h/°F.

A special case arises for the counter-flow HX when the inlet heat capacity rates are equal. In this situation the effectiveness equation reduces to the following:

$$\varepsilon = \frac{\mathrm{NTU}}{1+\mathrm{NTU}}\,.$$

Now the effectiveness and the maximum possible heat transfer rate can be related with the following equations to calculate the actual heat transfer within the exchanger:

$$q_{\max} = C_{\min}(T_{h,i}-T_{c,i})\,,$$

$$q = \varepsilon\, q_{\max}\,,$$

where,

$q_{\max}$ = maximum possible heat transfer rate, Btu/h

q = actual heat transfer rate, Btu/h

$T_{h,i}$ = hot fluid inlet temperature, °F

$T_{c,i}$ = cold fluid inlet temperature, °F.

It is now a simple matter to determine the outlet temperatures from the overall energy balances.

It should be noted that the above equation for the cross-flow HX type can be used with confidence only when $C_r \approx 1.0$. Therefore, the routine will warn the user with a message to the screen if C_r is greater than 1.15 or less than 0.85 for the cross-flow geometry.

The above methods of solution may be generalized to include other flow geometries via proper modification of the effectiveness equation. The derivations of the above effectiveness relationships are provided in references 1 and 2.

Pressure drop through the LLHX is found by conventional pressure drop calculation methods utilizing user-supplied equivalent lengths and equivalent diameters. See section 8.2 for a detailed explanation of the pressure drop calculations.

References

1. Incropera, F.P., and DeWitt, D.P.: "Fundamentals of Heat Transfer." Wiley, 1981.

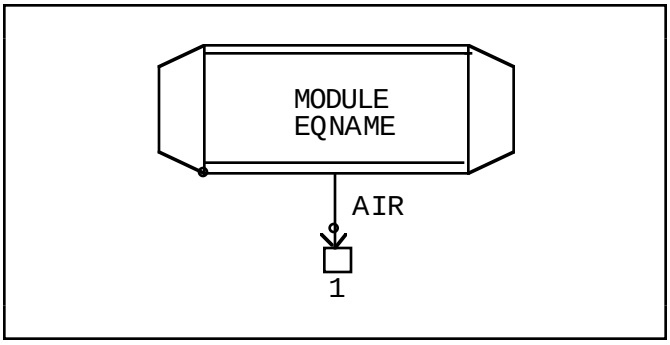
2. Kays, W.M., and London, A.L.: "Compact Heat Exchangers." McGraw-Hill, 1984.

Table 50. LLHX edit screen.

```
CASE NAME: DEFAULT            LLHX EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                   COMPONENT:                 RECORD NUMBER: 000001
_____________________________________________________________________________
|            ** GENERAL INPUT DATA **              |   ** RESOURCE TRACKING **  |
|10 OVERALL UA                (BTU/HR/F)  350.00   |                            |
|15 FLOW: 1=CNTR,2=PARAL,3=CROSS            1.00   |7 POWER   (WATTS)    0.00   |
|                                                  |8 WEIGHT    (LBM)    0.00   |
|         ** PRESSURE DROP PARAMETERS **           |9 VOLUME   (FT^3)   0.000   |
|11 SIDE A EQUIV LENGTH             (FT)    1.00   |                            |
|12 SIDE A EQUIV DIAMETER           (IN)   1.000   |                            |
|13 SIDE B EQUIV LENGTH             (FT)    1.00   |                            |
|14 SIDE B EQUIV DIAMETER           (IN)   1.000   |                            |
|                                                  |                            |
|                                                  |                            |
|                                                  |                            |
|__________________________________________________|____________________________|
|                ** BENCHMARK DATA **              |      ** OUTPUT DATA **     |
|                     MIN        NOM        MAX   |                            |
| HEAT TRANS   (W)  0.00E+00   0.00E+00   0.00E+00|17 HEAT TRANSFR (W)    0.00 |
|__________________________________________________|____________________________|
```

10.27 MODULE—MODULE Simulation Description

This subroutine may be used to simulate a module. It is designed to track the respirable atmospheric composition, mass additions/losses, and heat transfer to and from other modules, equipment and the external orbital environment. The equipment loads are determined by assigning subsystems to a module through the "ASSIGN" command. The effective sink temperatures may be timelined through the use of the OPS block logic.



10.27.1 Comparison to the CABIN Module

This routine is an update to the original CABIN routine. It has more detailed energy balance equations and thermal and mass transfer networks. The avionics air system is modeled as four rack banks, each having six racks lumped into one group. The air volume has been divided into three diffusion nodes that exchange flow as follows. The main air node for external leakage and exchange with other components is the aisleway node.

10.27.2 General Description

The major assumptions used in the current formulation of the module routine include the following:

A. Interior mean radiant and structure temperatures are equal.

B. Mixing of constituents is instantaneous and uniform over the module volume.

C. Particle concentrations and temperatures exit the volume at the calculated perfectly mixed conditions.

D. Energy loss due to module leakage is negligible.

The following sections provide a brief technical discussion of the mass and energy balance relationships utilized in modeling this component.

10.27.3 Energy Balance

Determination of the bulk air temperature involves the application of the generalized conservation of energy equation (open system formulation) in which the kinetic and potential energy terms are neglected. The equation is expressed as follows:

$$\dot{Q} + \sum \dot{m}_i h_i = \sum \dot{m}_e h_e + \left.\frac{dE}{dt}\right|_{\text{module}} .$$

Implementing the above assumptions and considering air to behave like an ideal gas, the bulk air energy equation may be finite differenced into an algebraic equation. In its indexed form it is expressed as follows:

$$T_i^{P+1} = T_j + \frac{\sum G_{ij} T_j^P + Q_i}{\varepsilon \sum G_{ij} + C_i/\Delta t}, \qquad (1)$$

where,

T_i^{P+1} = implicitly calculated temperature of $i^{\underline{th}}$ module at end of time step, °F

T_i^P = temperature of $i^{\underline{th}}$ module at beginning of time step, °F

G_{ij} = conductance of adjoining nodes, i.e., $\dot{m}C_p$, $kA/\Delta X$, etc., Btu/h °F

C_i = capacitance of $i^{\underline{th}}$ module, Btu/h °F

Q_i = impressed heating rate on $i^{\underline{th}}$ module, Btu/h

Δt = time step, hours

$$\varepsilon = \frac{\Delta t - C_i/\sum G_{ij}\left(1 - e^{\frac{\Delta t}{C_i/\sum G_{ij}}}\right)}{\Delta t\left(1 - e^{\frac{\Delta t}{C_i/\sum G_{ij}}}\right)} .$$

Similar equations are utilized for other module nodes (i.e., structure, shell). A schematic of the module thermal network is shown in figure 64.

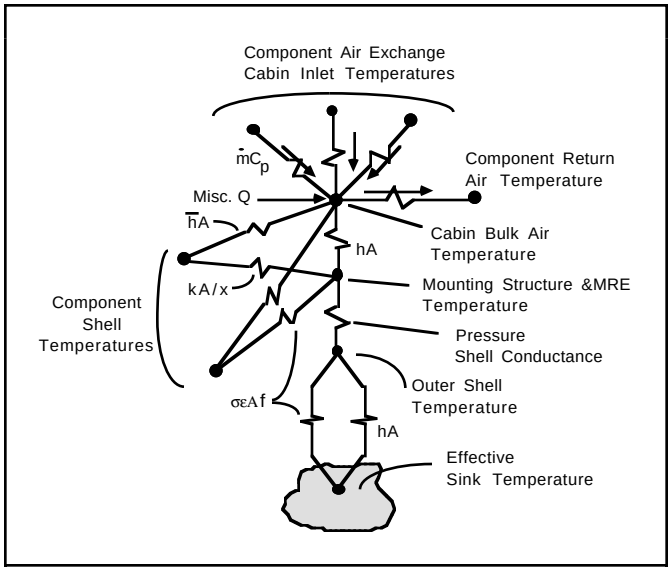


Figure 64. MODULE thermal network.

The module bulk air and structure nodes are thermal diffusion nodes. The thermal capacitance of the pressure shell, meteoroid shield, attachments, etc., is assumed to reside in the structure node. The module air mass, as calculated by the mass continuity equation, provides for the determination of the bulk air capacitance. The outer structure node is arithmetic and the effective environmental sink temperature is considered to be a thermal boundary. Sink temperatures may be timelined as a function of orbital period and included in the supplemental OPS logic for a given specific case. Orbital heating loads may also be imposed on the module air directly through Q_i (CLOAD array) in supplemental OPS logic.

Equipment heat rejection to the module will occur via one or all of the following mechanisms. Flow enthalpy transfers to and from the module are iteratively transferred by the C and PRO arrays to the inlet/outlet streams of the module. Component convective loading on the bulk air node is communicated via the CLOAD array and is based on the delta temperatures that exist between the equipment shell and module air. Mechanical conduction is allowed to occur between the component and module structure node. Radiative exchange between equipment and module is also provided using a mean radiant environment approach. Heat transfer rates for radiation and conduction are based on the delta temperatures existing between component shell and the solid structure node. These quantities are passed from the component routines to the module routine by the CLOAD array.

The implicit system of algebraic equations are solved by an iterative, block substitution technique. Internal to the routine a check is made on numerical integration stability based on the system time step and given conditions.

10.27.4 Mass Balance

The total atmospheric mass balance is performed by a flow balance on the participating individual species. The rate of constituent mass storage is governed by the conservation of mass equation and is given in its finite differenced (Euler) formulation as:

$$m_i^{P+1} = m_i^P + \Delta t\,(\sum \dot{m}_i^{\text{out}}), \qquad (2)$$

and the total atmospheric mass is given by,

$$m_t = \sum_j m_j ,$$

where;

m_i^{P+1} = mass of $i^{\underline{th}}$ constituent at end of time step, lbm

m_i^P = mass of $i^{\underline{th}}$ constituent at beginning of time step, lbm

$\sum \dot{m}_i^{\text{out}}$ = flow of $i^{\underline{th}}$ constituent into and out of module, respectively, lbm/h.

Incrementing of the module air mass per the explicit formulation given above will occur after the system relaxation criteria has been achieved for the current time step. Numerical integration stability of the Euler formulation is dependent on the time step choice. For the most part, diminishing time steps will give correspondingly greater accuracy to the approximation (at the expense of run time). It is therefore left to the user to determine a suitable system time step.

10.27.5 Pressure Calculations

The module pressure is determined from the ideal gas laws and may vary over the given time step. The pressure is updated every iteration and includes the effects from module leakage, EVA, and experiment vent loses. Module relief venting is determined at the completion of the time step in the 300 block.

10.27.6 Input Parameters

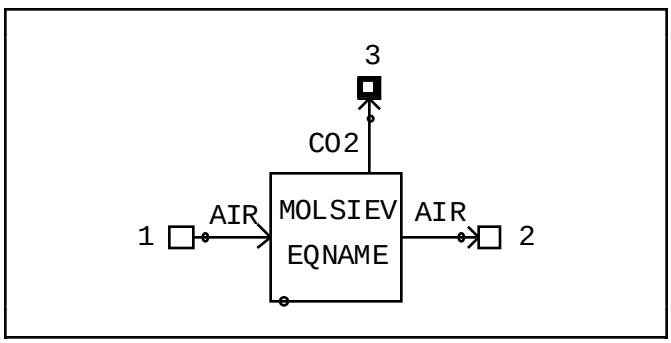
The input parameters are shown in table 51. The user must specify the module volume, structure mass and initial partial pressures, total pressure, and temperatures. An input is provided for the module pressure relief valve set point and for the experiment vent, EVA, and module leakage losses. These are specified as LBM/Day. Transient vent operations can be simulated through the OPS blocks by resetting the flowrates to the 24-h equivalent flowrate for the period in question. The module heat load input is used to simulate miscellaneous loads such as the module lighting but can be used to simulate environmental loads if required. If environmental heat loads are being imposed through the module heat load input parameter the heat transfer to the module sink should be adjusted accordingly. The interface to the external environment is simulated through the shell and external radiation conductances. The shell conductances should include any insulation present such as MLI. An external convection conductance is provided to simulate modules in a terrestrial environment.

Table 51. MODULE edit screen.

```
CASE NAME: DEFAULT          MODULE EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                 MODULE NAME:                  RECORD NUMBER: 000001
_____________________________________________________________________________
|10 MODULE LEAKAGE (LBM/DAY)     0.0 |20 AVIONICS SUPPY TEMP        (F)  70.00 |
|14 RELIEF SET POINT  (PSIA)   15.00 |21 AVIONICS TOTAL AIRFLOW   (CFM)  693.0 |
|11 AISLEWAY VOLUME   (FT^3) 1125.00 |22 AVIONICS AIR LEAKAGE (% TOTAL)  15.00 |
|12 STANDOFF VOLUME   (FT^3)  500.00 |23 STANDOFF TOTAL AIRFLOW   (CFM) 280.00 |
|13 AVIONICS VOLUME   (FT^3)  500.00 |24 INTERIOR PRESS SHELL TEMP  (F)  60.00 |
|15 INITIAL AIR TEMP     (F)  70.000 |25 AVIONICS HEAT LOAD        (W) 4000.00 |
|16 INITIAL AIR PRESS (PSIA)  14.700 |26 AISLEWAY HEAT LOAD        (W)    0.00 |
|17 INITIAL PPO2      (PSIA)    3.00 |27 STANDOFF HEAT LOAD        (W)    0.00 |
|18 INITIAL PPCO2    (MM HG)    3.00 |28 EQUIPMENT TOTAL MASS    (LBM) 10.0000 |
|19 INITIAL REL HUM     ( % )  40.00 |29 CONVECTION FLOW VELOCITY(FT/S)    1.0 |
|                                    |                                         |
|____________________________________|_________________________________________|
|                ** BENCHMARK DATA **            |       ** OUTPUT DATA **      |
|                        MIN      NOM      MAX  |                              |
| TEMPERATURE      (F)   0.000    0.000    0.000|30 CABIN TEMP     (F)   0.000 |
| TOTAL PRESS   (PSIA)   0.000    0.000    0.000|31 TOTAL PRESS (PSIA)   0.000 |
| PPO2          (PSIA)   0.000    0.000    0.000|32 PPO2        (PSIA)   0.000 |
| PPCO2        (MM HG)   0.000    0.000    0.000|33 PPCO2       (PSIA)   0.000 |
| REL HUMIDITY     (%)   0.000    0.000    0.000|34 PPH2O       (PSIA)   0.000 |
| DEW POINT        (F)   0.000    0.000    0.000|                              |
|_________________________________________________|______________________________|
```

10.28 MOLSIEV—Molecular Sieve Simulation Description

The molecular sieve (MOLSIEV) routine simulates the transient CO_2 removal characteristics of a Zeolite (4A, 5A, or 13X) molecular sieve. The theoretical approach to modeling the process is based on the determination of the overall CO_2 mass transfer rate as primarily a function of the equilibrium isotherm data. Since the process is cyclic by nature, the routine is coded such that time-averaged flows and cabin heat loads are computed if the system time step is not compatible with the user-specified half-cycle absorb/desorb durations. Detailed, two-bed simulation requires OPS control logic to cycle air flow to the active molecular sieve at the proper time. Currently, the routine is configured for a thermal-vacuum CO_2 desorb simulation. The component MOLSIEV, therefore, supplies (internal to the routine) a vacuum pump segment and thermostatic temperature controller logic.



10.28.1 General Assumptions

The following general assumptions were made in deriving the equations that are employed in the MOLSIEV routine.

A. The temperature gradient within the adsorbent pellets are negligible.

B. For the analysis of mass storage characteristics of a packed bed, it is assumed that gas and bed characteristics are uniform over any cross section of the bed normal to the gas flow direction, excluding intraparticle concentration gradients; i.e., no channeling of the flow.

C. Water poisoning effects are ignored.

D. Adsorption occurs by the diffusion of an adsorbate through the stagnant surface film at the exterior surface and then diffusion into the interior of the particle.

E. Heats of adsorption/desorption are independent of temperature and concentration.

F. In order to overcome small allowable stability time computing intervals, the pressure gradients within the bed during desorb are neglected. Since these gradients are relatively small this assumption is appropriate for most analyses.

The adsorption/desorption processes are modeled by a mass transfer network. There are three mass transfer mechanisms: (1) gas to adsorbent particle surface, (2) intraparticle diffusion, and (3) interparticle diffusion. A discussion of each is included in the following paragraphs.

10.28.2 Mass Transfer to Adsorbent Surface

The adsorbate gas can be transferred only between the bed surface and the carrier gas. CO_2 vapor densities are computed as a function of the inlet CO_2 concentration, packed bed mass transfer convective effectivity, and the characteristic CO_2 equilibrium vapor density of the particle surface.

A vapor density network is utilized as shown in figure 65 for the calculation of gas to adsorbent surface CO_2 mass transfer. The mass transfer convective effectivity is computed as follows:

$$\varepsilon_m = 1 - e^{-\frac{kA}{\dot{v}}} ,$$

where,

k = overall packed bed convective mass transfer coefficient, $\frac{\text{ft}}{\text{h}}$

A = zeolite total particle surface area, ft^2

$\dot{v}$ = volumetric flow rate of air, $\frac{\text{ft}^3}{\text{h}}$.

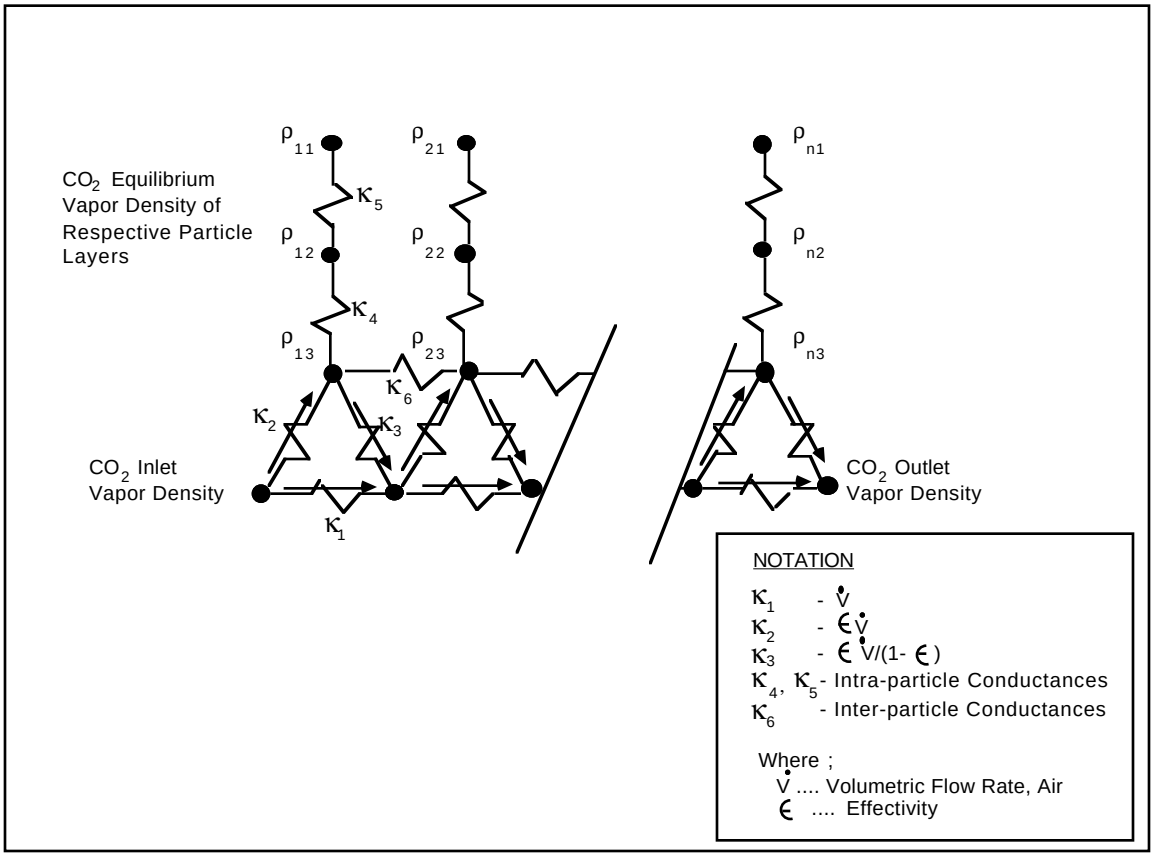


Figure 65. MOLSIEV mass transfer network.

The packed bed outlet CO_2 concentration calculation is given by:

$$\rho_{CO_2}^{out} = \frac{\rho_{CO_2}^{in} - \left[\varepsilon_m/(1-\varepsilon_m)\right]\rho_{eql}}{1 + \dfrac{\varepsilon_m}{1-\varepsilon_m}},$$

where,

ε_m = mass transfer convective effectivity

$\rho_{CO_2}^{in}$ = inlet CO_2 vapor density, lbm/ft^3

$\rho_{CO_2}^{out}$ = outlet CO_2 vapor density, lbm/ft^3

ρ_{eql} = bed equilibrium vapor density, lbm/ft^3.

10.28.3 Intraparticle Mass Transfer

The large thermal and mass concentration transients inherent in the adsorption/desorption processes suggest that an adsorbate gradient exists within the individual adsorbent particles. This is caused by an intraparticle diffusion resistance that, at least partially, governs the overall rate of mass transfer to/from the particle. The intraparticle mass transfer mechanism is analogous to heat transfer with a driving potential of equilibrium vapor density instead of temperature:

$$\dot{m}_{CO_2} = k(\rho_i - \rho_j) ,$$

where,

k = diffusion mass transfer conductance, ft^3/h

ρ_i = equilibrium vapor density of node i (lbm/ft^3)

ρ_j = equilibrium vapor density of node j (lbm/ft^3)

$\dot{m}_{CO_2}$ = rate of mass transfer from node i to node j (lbm/h)

To calculate the equilibrium vapor density for each particle layer:

$$\rho_i = \frac{\left(\frac{C_1}{dt}\right)(\rho_{i(o)}) + \sum_{j=1}^{n} K_{ij}\rho_j - \left(\frac{C_2}{\mathrm{dt}}\right)(T_i - T_{i(o)}) - \dot{m}_{im\rho}}{\left(\frac{C_1}{dt}\right) + \sum_{j=1}^{n} K_{ij}},$$

where,

$$C_1 = \frac{M_b}{R(T+460)} \left.\frac{\Delta f}{\Delta \rho}\right|_T$$

$$C_2 = M_b, R(T+460) \left.\frac{\Delta f}{\Delta T}\right|_{PP}$$

K_{ij} = conductance between layer i and adjacent layer j $\left(\frac{\mathrm{ft}^3}{\mathrm{h}}\right)$

$\dot{m}_{im\rho}$ = mass flow imposed on layer from adjacent elements $\left(\frac{\mathrm{lbm}}{\mathrm{h}}\right)$

ρ_i = equilibrium vapor density of layer i $\left(\frac{\mathrm{lbm}}{\mathrm{ft}^3}\right)$

$\rho_i(o)$ = equilibrium vapor density of layer i at end of last time step $\left(\frac{\mathrm{lbm}}{\mathrm{ft}^3}\right)$

T_i = temperature of layer i (°F)

$T_i(o)$ = temperature of layer i at end of the last time step (°F)

n = total number of adjacent nodes

M_b = adsorbent bed mass (lbm)

C_1, C_2 = intraparticle mass transfer capacitances.

10.28.4 Interparticle Mass Transfer

During the adsorb half cycle, it is likely that the gas-to-particle mass transfer governs the axial movement of adsorbate down the bed and that the interelement mass transfer can be ignored. However, the thermal vacuum desorb half-cycle flow down the bed is relatively low and the inter-element conductance may become a significant factor. The mass transfer between elements is calculated by:

$$\dot{m}_{ij} = Kij(\rho_i - \rho_j),$$

where,

$\dot{m}_{ij}$ = mass transfer from element i to element j $\left(\frac{\mathrm{lbm}}{\mathrm{h}}\right)$

ρ_i, ρ_j = equilibrium vapor density of surface layer of element i and j, respectively.

10.28.5 Mass Balance (Desorb)

In the desorb half-cycle, the absence of a carrier gas and the forcing function of vacuum pump volumetric flow would require a large number of bed elements and corresponding computer time, and is impractical for a system level model. Because of the relatively low flow rate, the bed length effects are much less significant than in the adsorb cycle. This allows a modified approach where the bed surface layer partial pressures are averaged and used as the inlet pressure for the vacuum pump. Volumetric flow calculations are as follows:

$$\dot{v} = CDSP\left[1 + CLF\left(1 - \left(\frac{P_{acm}}{P_{\mathrm{avg}}}\right)^{\frac{1}{2}}\right)\right],$$

where,

$CDSP$ = vacuum pump piston displacement $\frac{\mathrm{in}^3}{\mathrm{rev}}$

CLF = clearance factor

P_{avg} = average surface pressure of bed particle (lb/in^2 absolute)

P_{acm} = downstream accumulator pressure (lb/in^2 absolute)

n = polytropic exponent.

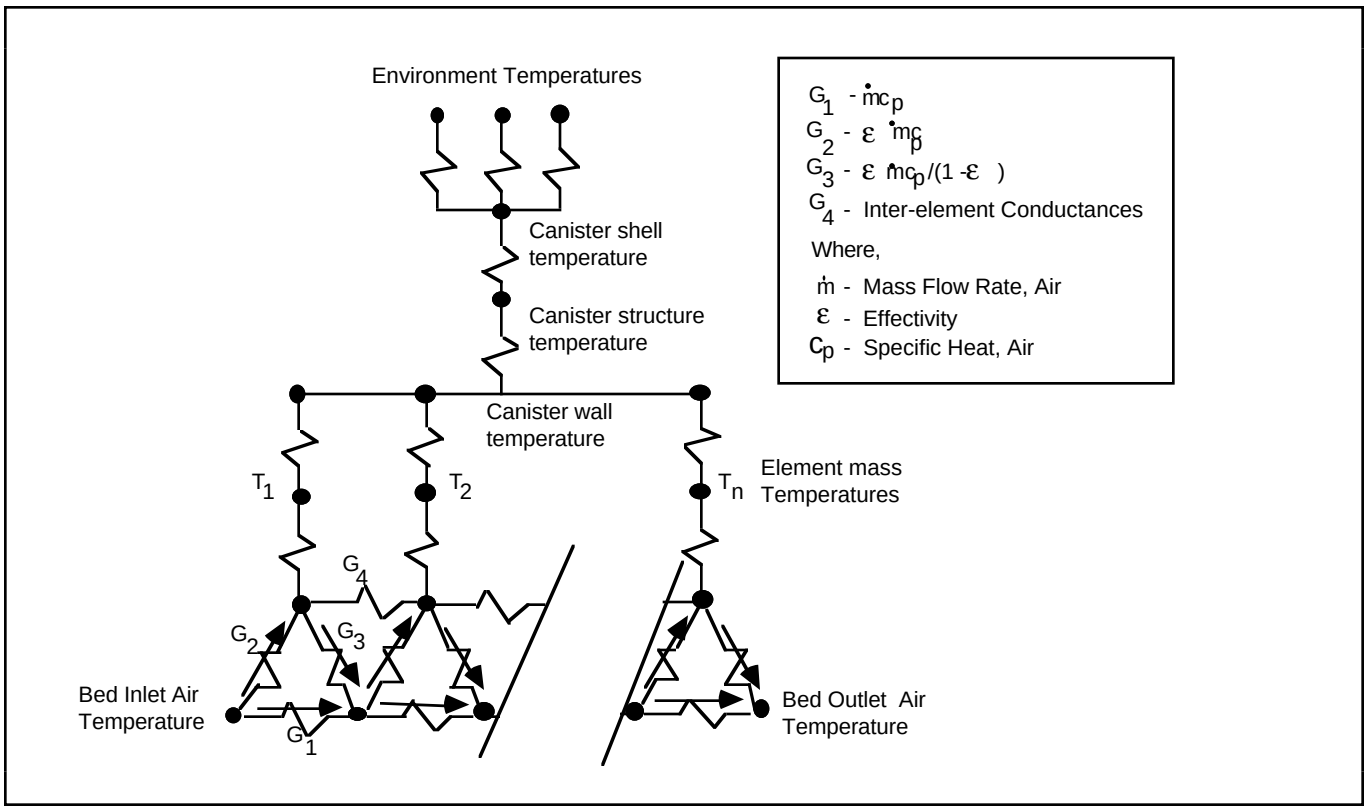


Figure 66. MOLSIEV thermal network.

10.28.6 Energy Balance (Adsorb)

An R-C network approach is utilized for the determination of the zeolite bed, canister, outer insulation, and assembly free stream outlet temperatures. The thermal network is shown in figure 66.

The convective delta technique is utilized to simulate the fluid-to-wall heat transfer mechanisms that typically dominate the overall heat transfer rates in a packed bed. Calculation of the thermal delta network effectivity is analogous to the mass transfer effectivity calculation and is expressed as:

$$\varepsilon_t = i - e^{-\left(\frac{hA}{\dot{m}Cp}\right)},$$

where,

h = overall packed bed convective heat transfer coefficient $\frac{\text{Btu}}{\text{h}\bullet\text{ft}^2\bullet\text{F}}$

$\dot{m}$ = mass flow rate of air $\left(\frac{\text{lbm}}{\text{h}}\right)\dot{m}$

Cp = specific heat of air $\frac{\text{Btu}}{\text{lbm}\bullet\text{F}}$

A = total surface area of bed particles, ft^2.

Each bed element is tied to upstream and downstream elements as well as to the canister shell by linear conductors. The following equation is utilized to calculate the temperature of each diffusion node:

$$T_i = \frac{\left(\frac{C}{dt}\right)T_i(o) + \sum_{j=1}^{n} G_{ij}T_j + \dot{Q}_{imp}}{\frac{C}{dt} + \sum_{j=1}^{n} G_{ij}},$$

where,

T_i = temperature of node i (°F)

$T_i(o)$ = temperature of node i at end of last time step (°F)

G_{ij} = thermal conductance between node i and j $\left(\frac{\text{Btu}}{\text{h•F}}\right)$

C = thermal capacitance $\left(\frac{\text{Btu}}{°\text{F}}\right)$

dt = time step (h)

$\dot{Q}_{imp}$ = imposed heat flow such as heat of adsorption $\left(\frac{\text{Btu}}{\text{h}}\right)$

n = total number of adjacent nodes.

The arithmetic node temperatures are calculated by:

$$T_i = \frac{\sum_{J=1}^{n} G_{ij} T_j}{\sum_{J=1}^{n} G_{ij}} .$$

10.28.7 Energy Balance (Desorb)

During the desorb half-cycle of the MOLSIEV, an additional delta network is included using TNETWK to model the heat loss through the vacuum pipe between the bed outlet and the vacuum pump inlet. The inputs and outputs for this subroutine call are similar to those of the bed thermal network.

High temperature simulation of the sorption bed is achieved via thermostatic temperature controller logic. Proportional control logic is used to determine the relative magnitude of the impressed heat rate to obtain and hold the desired set point temperature. Temperature controllers and vacuum pipe conductance constants are user addressable.

10.28.8 Pressure Drop

Pressure drop through the MOLSIEV during the adsorb half-cycle is calculated based on user-specified equivalent lengths and diameters in the subroutine PIPEDP. The pressure drop in the desorb half-cycle is calculated in the PSPEC subroutine.

10.28.9 Input Parameters

The input parameters for the MOLSIEV routine are shown in table 52. The output and benchmark data is shown in table 53.

References

1. "Verification Analysis of the Computer Aided System Engineering and Analysis (CASE/A) Molecular Sieve (MOLSIEV) and Desiccant Bed (DEFLOW) Component Routines," MDC W5183.

2. "Computer Analysis Techniques for Spacecraft Adsorption Beds," A69-21931.

3. Benet and Myers: "Momentum, Heat, and Mass Transfer." McGraw-Hill Book Company, 1954, pp. 509, 510, 511.

4. McAdams: "Heat Transmission." Third edition, McGraw-Hill Book Company, 1962, pp. 294, 295, 296.

5. "Carbon Dioxide Adsorption Isotherm Data Sheet No. 22, Molecular Sieve Type 5A Pellets," Union Carbide.

6. "A Transient Performance Method for CO_2 Removal With Regenerable Adsorbents," NASA CR-112098.

Table 52. MOLSIEV main edit screen.

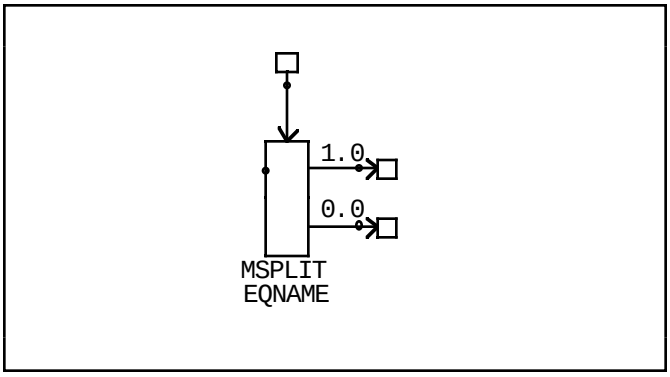
```
CASE NAME: DEFAULT                                  MOLSIEV EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                              ** GENERAL INPUT DATA **                                  |          ** THERMAL CHARACTERISTICS **          |
| 10 BED FLAG     (1-5A,2-EMRC)    1.00 | 12 HEAT OF ADSRBTN  (BTU/LB-CO2)  450.00 |_________________________________________________|
| 11 ADSORBENT BED WEIGHT (LBM)    8.00 | 20 BED-GAS MAS TRNS COEF (FT/HR)  980.00 |          ** BED THERMAL CONDUCTANCES **         |
| 27 PARTICLE SURF AREA  (FT^2)  23.000 | 58 INTRA-PART MAS TRNS G(FT^3/H) 1125.00 | 50 INTER-ELMT THERMAL G    (BTU/HR/F)    1.00 |
| 36 CANISTER WEIGHT      (LBM)  10.000 | 59 INTER-ELMT MAS TRNS G(FT^3/H)    0.00 | 19 BED-GAS HT TRANS COEF(BTU/FT^2/HR)    4.50 |
| 38 NUMBER OF BED ELEMENTS         6.0 |** VAC PUMP/COMPRESSOR PERFORMANCE DATA **| 51 BED-CANN  CONDUCTANCE   (BTU/HR/F)   25.00 |
| 14 BED LOAD CAPACITY FACTOR      1.00 | 30 VACUUM PUMP DISP    (FT^3/HR)  115.00 | 24 CANN INSUL CONDUCTANCE  (BTU/HR/F)    1.00 |
|                                       | 31 CLEARANCE FACTOR               0.0200 | 56 VAC PIPE THERMAL EFFECTIVITY         0.999 |
| 40 HEATER FLAG   (0-OFF,1-ON)      0. | 32 POLYTROPIC EXPONENT            1.1100 | 52 VAC PIPE THERMAL CAPACIT  ( BTU/F)    0.10 |
| 41 VAC PMP FLAG  (0-OFF,1-ON)      0. |   ** VACUUM VENT TO SPACE PARAMETERS **  |        ** ENVIRONMENT CONDUCTANCES **           |
| 15 OPS MODE FLAG (0=ADS,1=DES)     0. | 42 LINE ID                  (IN)  0.0000 |                      CANNISTER      VAC PIPE |
| 29 CONTROLLER GAIN              0.100 | 43 LINE EQ LENGTH CONTINUUM (FT)    0.00 | CONVECT (BTU/H/F) 16     0.20   53    20.000 |
| 13 BAKEOUT SETPOINT TEMP  (F)  400.00 | 44 LINE EQ LENGTH MOLECULAR (FT)    0.00 | STRUCT  (BTU/H/F) 17     3.00   54     2.500 |
| 28 MAX HEATER POWER   (WATTS)  575.00 | 45 SPACE LINK PRESSURE   (TORR)  0.0E+00 | RAD FAE    (FT^2) 18     4.00   55     7.500 |
|                                       |      ** PRESSURE DROP PARAMETERS **      |             ** SOLUTION CONTROL **              |
|       ** INITIAL CONDITIONS **        | 22 AIR STREAM EQUIV LENGTH (FT)     1.00 | POWER (WATTS)  100.00  35 MAX ITERS      50.0 |
| 26 INIT BED TEMP          (F)   65.00 | 23 AIR STREAM EQUIV DIA     (IN)    6.00 | WEIGHT  (LBM)   50.00  34 DMPNG FCT    0.0000 |
| 39 INIT BED LOAD        (LBM) -0.3355 |** OUTPUTS & BENCHMARKS ** (^L TO OPEN) _ | VOLUME (FT^3)    2.00  33 TEMP RLX(F) 0.00001 |
|_______________________________________|_______________________________________|_______________________________________________|
```

Table 53. MOLSIEV output edit screen.

```
CASE NAME: DEFAULT          MOLSIEV OUTPUTS EDIT SCREEN    LAST UPDATED: 890215
SUBSYSTEM:                      COMPONENT:                RECORD NUMBER: 000001
_____________________________________________________________________________________________
|                                ** OUTPUT DATA **                                          |
|                                                                                           |
| 62 **OPEN OUTPUT**                 0.0  63 TOTAL CO2 REMOVED (LBM)      0.000             |
| 61 MAXIMUM CO2 BED LOADING(%)    0.00  64 FINAL CO2 BED LOADING(LBM)   0.000             |
|                                                                                           |
|___________________________________________________________________________________________|
|                              ** BENCHMARK DATA **                                         |
|                                                                                           |
|                                 MIN            NOM           MAX                         |
|        AIR IN   (LBM/HR)   66   0.000     67   0.000    65   0.000                       |
|        AIR OUT  (LBM/HR)   69   0.000     70   0.000    68   0.000                       |
|        CO2 OUT  (LBM/HR)   72   0.000     73   0.000    71   0.000                       |
|___________________________________________________________________________________________|
```

10.29 MSPLIT—Multisplit Simulation Description

The MSPLIT routine provides a multiple flow split function. The current routine can accommodate up to seven outlet streams. The outlet streams all have the same fluid composition and temperature. The MSPLIT is an adiabatic device.



The location format of the MSPLIT device is as follows:

```
LO;MSPLIT;EQNAME;NLEGS;Fraction 1;
   Fraction 2•••Fraction NLEGS
```

where NLEGS is the number of outlet legs (up to seven) and fraction 1 through fraction NLEGS are the split fractions for the amount of inlet flow to be sent through each leg.

10.29.1 General Discussion

In this routine, the user can either set the split fractions to given numbers or specify the split fractions as zeroes. In the first method, the user must ensure that the fractions entered sum to 1.0. The incoming mass flow will then be split according to the user provided fractions. If the latter method is used, the flow splits will be calculated by the routine according to the individual leg downstream resistances. In this way, the MSPLIT routine may be used as a flow balancing device.

10.29.2 Mass Calculations

The flow balance calculations are governed by the system damping factor specified by the user in the CONTROL data base. The new flow rates for the outlet legs are given by the equation:

$$\dot{m}_j = \dot{m}_o + k(\dot{m}_b - \dot{m}_o) ,$$

where,

$\dot{m}_j$ = new flow rate of outlet j

$\dot{m}_o$ = old flow rate of outlet j

$\dot{m}_b$ = balanced flow rate of outlet j

k = system damping factor.

This method is very similar to that utilized in the NODE calculations and the user should refer to section 10.30 for further discussion.

10.29.3 Pressure Calculations

The pressure of the MSPLIT is calculated according to the following equation:

$$P_S = \frac{\sum_{j=1}^{k} G_j P_j}{\sum_{j=1}^{k} G_j} ,$$

where,

P_s = pressure of the MSPLIT (lb/in^2 absolute)

P_j = pressure of connecting stream j (lb/in^2 absolute)

G_j = flow conductance of connection j (lbm/h/lb/in^2 absolute)

k = total number of streams connected to the MSPLIT.

The flow conductance of each connection is given by the quasi-linear approximation:

$$G_j = \frac{\dot{m}_j}{\Delta P_j} ,$$

where,

ΔP_j = the pressure drop for connection j (lb/in^2 absolute)

The connection pressure drops are calculated by the solution system routines discussed in section 8.2. The MSPLIT routine will automatically send the split fractions based on the hydraulic configuration of the downstream component and the K factors of each leg if the initial split fractions are set to 0.0 by the user. This option allows the MSPLIT to act as a variable flow adjustment device based on the hydraulic balance solution.

10.29.4 Input Parameters

The data base parameters for the MSPLIT routine are shown in table 54. The optional hydraulic K factor for each outlet leg gives a pressure drop for the corresponding outlet leg according to:

$$\Delta\rho_{\text{leg}} = K\dot{m}^2 ,$$

where,

$\Delta\rho_{\text{leg}}$ = pressure drop for outlet leg, lb/in^2 absolute

K = flow resistance coefficient

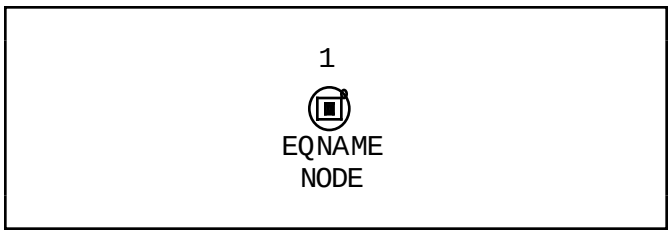
$\dot{m}$ = mass flow rate through leg, lbm/h.

Table 54. MSPLIT edit screen.

```
CASE NAME: DEFAULT                                  MSPLIT EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:              COMPONENT:                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                              ** GENERAL INPUT DATA **                                |            ** MISC PARAMETERS **          |
| 10 TOTAL NUMBER OF OUTLET STREAMS      2.0 | 18 FLOW SOLUTION DAMPING CONST 0.1000000 | 27 ** OPEN INPUT LOCATION **       0.000 |
|                                            |                                          | 28 ** OPEN INPUT LOCATION **       0.000 |
| 11 SPLIT FRACT OF OUTLET LEG #1    0.00000 | 19 HYDRAUL K FACT OF OUT LEG 1  0.00E+00 | 29 ** OPEN INPUT LOCATION **       0.000 |
| 12 SPLIT FRACT OF OUTLET LEG #2    0.00000 | 20 HYDRAUL K FACT OF OUT LEG 2  0.00E+00 | 30 ** OPEN INPUT LOCATION **       0.000 |
| 13 SPLIT FRACT OF OUTLET LEG #3    0.00000 | 21 HYDRAUL K FACT OF OUT LEG 3  0.00E+00 | 31 ** OPEN INPUT LOCATION **       0.000 |
| 14 SPLIT FRACT OF OUTLET LEG #4    0.00000 | 22 HYDRAUL K FACT OF OUT LEG 4  0.00E+00 |                                          |
| 15 SPLIT FRACT OF OUTLET LEG #5    0.00000 | 23 HYDRAUL K FACT OF OUT LEG 5  0.00E+00 |         ** RESOURCE TRACKING **          |
| 16 SPLIT FRACT OF OUTLET LEG #6    0.00000 | 24 HYDRAUL K FACT OF OUT LEG 6  0.00E+00 | 7 POWER                  (WATTS)    0.00 |
| 17 SPLIT FRACT OF OUTLET LEG #7    0.00000 | 25 HYDRAUL K FACT OF OUT LEG 7  0.00E+00 | 8 WEIGHT                   (LBM)    0.00 |
|                                            |                                          | 9 VOLUME                  (FT^3)    0.00 |
|____________________________________________|__________________________________________|________________________________________|
|                        ** BENCHMARK DATA **                                |                    ** OUTPUT DATA **                  |
|                          MIN         NOM         MAX                       |   36 OPER MODE,0=AUTO-SET 1=USER-SET       0.000          |
| INLET FLOW    (LBM/HR)    0.000       0.000       0.000                    |   33 PRESENT INLET FLOW     (LBM/HR)       0.000          |
| SPLIT PRESSURE  (PSIA)    0.000       0.000       0.000                    |   34 PRESENT INLET PRESSURE   (PSIA)       0.000          |
| FLOW TEMPERATURE   (F)    0.000       0.000       0.000                    |   35 PRESENT INLET TEMPERATURE   (F)       0.000          |
|____________________________________________________________________________|________________________________________________________|
```

10.30 NODE—NODE Simulation Description

The NODE component provides a general purpose flow balancing function based on the connecting stream pressures and flow resistances through the connections. It may act as either a flow summation device, a flow splitting device or a combination of both. A variable number of component streams may be connected to the NODE component up to a maximum number of 50. This number is a strictly arbitrary limit imposed by the size of the internal storage arrays in the routine and may be expanded easily if the need should arise. The NODE component may act as either an "active" or "passive" device with respect to mass flow, depending on the pressures of the connecting streams. The icon configuration is shown below.



10.30.1 General Assumptions

The solution of the pressure drop for the connection lines is based upon Darcy's formula that is applicable to all fluids. This formula provides a means of analyzing flow through conduits on the basis of equivalent length and equivalent diameter. The NODE thermal calculations are based on the assumption of reversible adiabatic mixing of the incoming flow streams with no chemical reactions.

10.30.2 Mass and Pressure Calculations

The NODE determines the flow rates and flow directions of the connecting streams based on their pressures, flow path resistances, and hydraulic classification codes to arrive at a mass balance for the NODE. The solution algorithm involves an implicit iteration on the flow rates and relies heavily on the pressure feedback functions of the system solution routines. An implicit algorithm is necessary due to the nature of the pressure and flow balancing process: the flow rates depend on the pressure differentials that in turn are dependent on the flow rates.

The routine first calculates the node pressure via an R-C network analysis technique according to the formula:

$$P_i = \frac{\sum_{j=1}^{n} G_{ij} P_j}{\sum_{j=1}^{n} G_{ij}} ,$$

where,

P_i = the node pressure (lb/in^2 absolute)

P_j = the connecting stream pressure (lb/in^2 absolute)

G_{ij} = the flow conductance between the node and stream j (lbm/h/lb/in^2 absolute).

Note the quasi-linear approximation below has been used for the conductance:

$$G_{ij} = \dot{M}_{ij} / \Delta P_j ,$$

where,

$\dot{M}_{ij}$ = the previous mass flow between stream j and the node (lbm/h)

ΔP_j = the pressure drop through the flow path to stream j (lb/in^2 absolute).

The pressure drop through each flow connection is calculated by the routine FRICTDP discussed in section 8.2. The flow rates and directions for each connecting stream that should occur for the stream-to-node pressure differentials and flow path dimensions are then calculated according to:

$$\dot{M}'_{ij} = G'_{ij} (P_i - P_j) ,$$

where,

$\dot{M}'_{ij}$ = the desired mass flow between stream j and the node (lbm/h)

G'_{ij} = the desired conductance between j and the node (lbm/h/lb/in^2 absolute)

These "desired" values are arrived at through an implicit iteration on the flow rate and conductance to achieve the differential between the connecting stream pressure and the node pressure calculated above. The desired flow rates are then altered by a damping factor to arrive at the damped flow rates for the present flow conditions:

$$\dot{M}''_{ij} = \dot{M}_{ij} + K(\dot{M}'_{ij} - \dot{M}_{ij}),$$

where,

$\dot{M}''_{ij}$ = the damped mass flow between stream j and the node (lbm/h)

K = the flow damping factor (user input, $0 < K < 1$).

The incoming and outgoing stream flows are then adjusted by a scaling factor to achieve a mass balance for the NODE. This is done by calculating the total amounts of the damped incoming and outgoing flows and then adjusting the appropriate stream flows to achieve an equal amount of incoming and outgoing flow. The total flow value to be scaled to is dictated by the user, via an input parameter in the NODE data base, as one of three choices:

A. The average value of the damped incoming and outgoing total flows (user flow flag = 0.0)

B. The damped incoming total flow (user flow flag = 1.0)

C. The damped outgoing total flow (user flow flag = 2.0).

The incoming final flows are then mixed together to form the fluid composition of the outgoing flow. It is assumed that no chemical reactions occur and that there are no fluid accumulation effects (the NODE has zero volume).

An appropriate choice of the user flow flag will speed the solution of complicated flow networks by biasing the NODE response toward a known flow-driving force (such as a PUMP component). For example, suppose a PUMP component drives the flow into a NODE component that splits the flow among multiple outlet branches. The best choice of the flow flag is 1.0, which causes the outgoing flow to be scaled to the incoming flow from the PUMP, since the PUMP flow will be a set quantity. The default value is 0.0.

The damping approach is necessary to stabilize the response of the NODE. The flow rates tend to become highly unstable for flow conditions that yield large flow conductances (i.e., gaseous flows). The damping factor should be set in the range of 0.01 to 0.5 for liquid flows and in the range of 1.0E-5 to 1.0E-3 for gaseous flows. These values are strictly rule of thumb, and the user is warned that experimentation is usually required to achieve an acceptable tradeoff between execution speed and stability. Also, the user must make sure that the damping factor is not set so low that it artificially causes convergence of the overall system solution by limiting the NODE flow or pressure variations to values below the system relaxation criterion. This problem is especially acute when gaseous flows are involved. Monitoring of the flow rates and pressures of crucial component streams for each overall system iteration pass is recommended for simulations that require very accurate flow and pressure balances. This may be accomplished via the OPS logic blocks or the interactive "print-to-display" command of the CNTRLLR component. Also, the user is warned that a large number of system iteration passes may be required (in the hundreds) if the NODE response dictates a very small damping factor.

10.30.3 Thermal Calculations

The NODE thermal calculations are based on the assumption of reversible adiabatic mixing of the incoming flow streams with no chemical reactions. The temperature of the incoming mixture is calculated according to:

$$Ti = \frac{\sum_{j=1}^{k} \dot{M}_j C_{pj} T_j}{\sum_{j=1}^{k} \dot{M}_j C_{pj}},$$

where,

T_i = the mixture temperature (°F)

T_j = the temperature of incoming stream j

$\dot{M}_j$ = the mass flow rate of stream j (lbm/h)

C_{pj} = the specific heat of stream j (Btu/lbm/°F)

10.30.4 Input Parameters

The NODE data base parameters and their default values are shown in table 55.

Table 55. NODE edit screen.

```
CASE NAME: DEFAULT              NODE EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                     COMPONENT:                   RECORD NUMBER: 000001
_____________________________________________________________________________
|                  ** INPUT DATA **                |   ** RESOURCE TRACKING **   |
|10 DAMPING FACTOR                        0.50000 |7 POWER    (WATTS)    0.00   |
|11 FLOW FLAG (0=AVG,1=IN,2=OUT)              0.0 |8 WEIGHT     (LBM)    0.00   |
|12 INITIAL PRESSURE              (PSIA)    14.70 |9 VOLUME    (FT^3)    0.00   |
|_____________________________________________|_____________________________|
```

10.31 O2N2—Oxygen/Nitrogen Control Panel Simulation Description

The O2N2 routine is used in conjunction with the CABIN routine in order to regulate cabin set point pressures. The unit senses changes in cabin pressure and regulates to the set point by adding makeup flow that is proportional to the set point offset. The proportional factor is input as a gain factor with the inlet flow rate being determined by the following relation:

Inlet Flow = Gain (set point - cabin pressure).

If the oxygen partial pressure falls below its nominal set point, the make up flow is from the oxygen tank, otherwise the make up flow is from the nitrogen tank. This mode of operation simulates the current Spacelab O_2N_2 control panel.

10.31.1 General Discussion

Power loads resulting from actuator control are dissipated directly into the cabin through the CLOAD array. The pressure of the outlet stream is assumed to be the same as that of the cabin. The gain factor should be chosen carefully. If the gain is set high then instabilities in the model simulation can be introduced. If instabilities do occur then the user should adjust the gain factor.

10.31.2 Input Parameters

The user input parameters for the O2N2 control panel routine are shown in table 56.

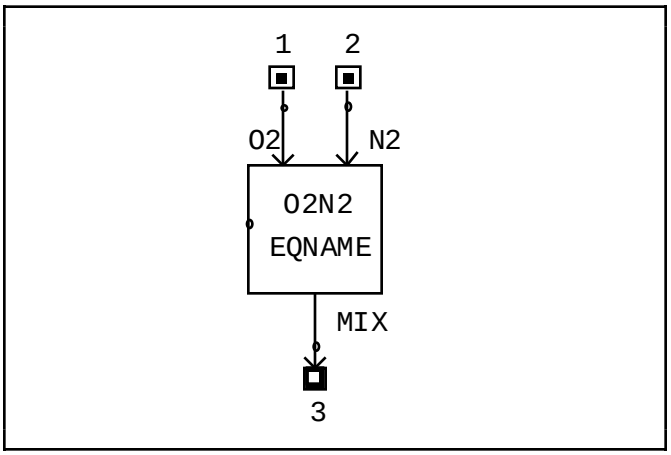
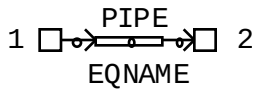


Table 56. O2N2 edit screen.

```
CASE NAME: DEFAULT              O2N2 EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                    COMPONENT:                  RECORD NUMBER: 000001
_____________________________________________________________________________
|                  ** INPUT DATA **                |   ** RESOURCE TRACKING **  |
|10 MAXIMUM O2 FLOW RATE        (LBM/HR)   13.00 |7 POWER    (WATTS)    0.00   |
|11 CABIN PPO2 PRESSURE SET POINT (PSIA)    2.00 |8 WEIGHT     (LBM)    0.00   |
|12 CABIN TOTL PRESSURE SET POINT (PSIA)   14.70 |9 VOLUME    (FT^3)    0.00   |
|13 MAXIMUM N2 FLOW RATE        (LBM/HR)  136.30 |                           |
|_____________________________________________________|___________________________|
|                            ** BENCHMARK DATA **                              |
|                      MIN        NOM        MAX                              |
| 02 SUPPLY (#/HR)   0.00E+00   0.00E+00   0.00E+00                            |
| N2 SUPPLY (#/HR)   0.00E+00   0.00E+00   0.00E+00                            |
|_____________________________________________________________________________|
```

10.32 PIPE—PIPE Simulation Description

The PIPE component routine simulates the heat loss and pressure drop characteristics of a smooth walled pipe. The component allows either one-way or bi-directional flow specification to simulate flow in either direction during a simulation. The PIPE component is a "passive" device with respect to mass flow—it only accepts the upstream mass flow or relies on a downstream component to pull flow through it. Therefore, an "active" flow generating component such as a PUMP must be located somewhere upstream or downstream of the PIPE to drive the flow through it. The icon configuration and stream definitions are shown below.



10.32.1 General Assumptions

The solution of the pressure drop for the PIPE is based upon Darcy's formula since it is a general formula that is applicable to all fluids. This formula provides a means of analyzing flow through the PIPE on the basis of equivalent length and equivalent diameter.

10.32.2 Mass Calculations

The direction of the fluid flow through the PIPE may be specified by the user as being one of three possible choices: (1) only from stream 1 to stream 2, (2) only from stream 2 to stream 1, or (3) conditionally determined by the routine itself depending on the connecting component stream pressures. The flow direction flag is set by the user as an input parameter in the PIPE data base. If the user sets the direction incorrectly for a given flow network, such as setting the flow direction opposite to the flow from the outlet of a PUMP, the solution system will recognize this condition and print a warning message to the screen. The user should always set the correct flow direction, if it is known, in order to speed the convergence of the overall system flow and pressure balance solutions, especially when NODE components or other flow balancing components are involved.

The fluid composition is merely passed from the functional inlet stream of the PIPE straight through to the outlet stream with no leakage loss or composition change.

10.32.2 Thermal Calculations

An R-C network analysis technique is utilized in the thermal calculations and is based on the generic network discussed in section 8.3 that allows for three modes of heat transfer with the environment: convection, conduction, and radiation. The network solution calculates the fluid outlet temperature and structural temperatures based on the user-specified conductances, thermal capacitance of the mass node, and environment and fluid inlet boundary temperatures. There are four linear conductances the user must provide as inputs along with the emissive area term for radiation and a fluid-to-wall heat transfer effectivity. Also, the user must specify an initial mass temperature, a relaxation criterion for all temperatures and a maximum iteration count not to be exceeded in the solution routine. The user should refer to the discussion in section 8.3 for detailed information on the network and the solution method involved.

10.32.4 Pressure Calculations

The PIPE routine utilizes a user-specified equivalent length and diameter to calculate the pressure drop across the pipe. The PIPE routine calls the generic solution routine FRICTDP (discussed in section 8.2) that calculates the pressure drop for the equivalent length method.

10.32.5 Input Parameters

The PIPE default data base parameters are shown in table 57.

Reference

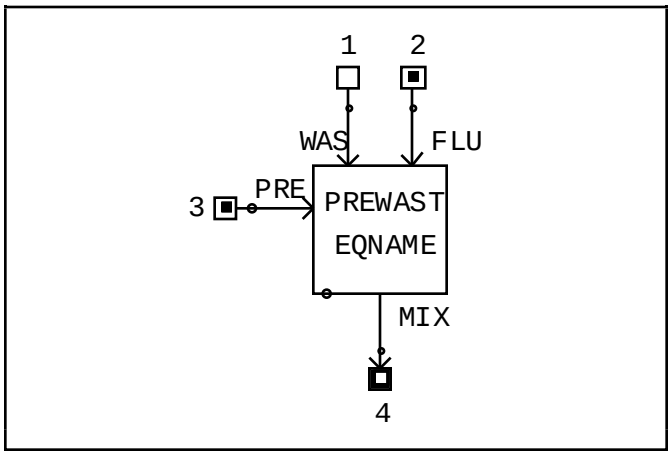
1. "ASHRAE HANDBOOK 1981 FUNDAMENTALS," American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc., 1981.

Table 57. PIPE edit screen.

```
CASE NAME: DEFAULT                                    PIPE EDIT SCREEN                                    LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                      RECORD NUMBER: 000001
_________________________________________________________________________________________________________________________________
|              ** GENERAL INPUT DATA **              |  ** RESOURCE TRACKING ** |          ** THERMAL INTERFACE DATA **          |
| 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)         1.0  | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   2.000  |
| 11 HYDRAULIC EQUIVALENT LENGTH       (FT)   10.00  | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.000  |
| 12 HYDRAULIC EQUIVALENT DIAMETER     (IN)    0.50  | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   2.000  |
| 13 HYDRAULIC K FACTOR    (K = Dp(X/M)**2)    0.00  |                          | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)    0.20  |
|                                                    |  ** SOLUTION CONTROL **  | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000  |
| 20 INITIAL MASS TEMPERATURE           (F)   75.00  | 22 RELAXATION    0.00001 | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00  |
| 21 THERMAL CAPACITANCE            (BTU/F)   1.000  | 23 MAX LOOP CNT    100.0 |                                                  |
|____________________________________________________|__________________________|__________________________________________________|
|                        ** OUTPUT DATA **                  |    ** BENCHMARK DATA **        MIN         NOM         MAX        |
|            25 PRESENT MASS TEMPERATURE     (F)    0.00    |    MASS TEMPERATURE      (F) 0.00E+00    0.00E+00    0.00E+00    |
|            26 PRESENT SHELL TEMPERATURE    (F)    0.00    |    SHELL TEMPERATURE     (F) 0.00E+00    0.00E+00    0.00E+00    |
|            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00    |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00    0.00E+00    0.00E+00    |
|____________________________________________________________|_____________________________________________________________________|
```

10.33 PREWAST—Waste Pretreatment Simulation Description

The PREWAST routine simulates a generalized waste treatment unit that simulates the consumption of water for waste flush and the introduction of water treatment additives. The amounts of the additive and flush water are based on the magnitude of the inlet waste stream 1 (which must have an active component such as a PUMP upstream) on a pound-for-pound basis. These values must be provided by the user.



10.33.1 General Discussion

Streams 2 and 3 should not have an active component upstream since they "pull" in flow based on stream 1. If either stream 2 or 3 is not used, a tank must be connected to the unused stream to avoid a run-time error.

10.32.2 Pressure Calculations

The waste pretreatment routine requires the user to input a pressure for the outlet mixture stream based on the assumption that the unit contains an integral fluid pump that will pressurize the outlet stream. The system pressure feedback function may be activated by setting the outlet pressure to zero.

10.33.3 Thermal Calculations

The temperature of the incoming mixture is calculated as the weighted average of the incoming stream temperatures:

$$T_o = \frac{\sum_{j=1}^{n} \dot{m}_j C_{p_j} T_j}{\sum_{j=1}^{n} \dot{m}_j C_{p_j}},$$

where,

T_o = mixture stream temperature (°F)

T_j = inlet stream j temperature (°F)

$\dot{m}_j$ = inlet stream j mass flow (lbm/h)

C_{p_j} = inlet stream j specific heat (Btu/lbm/°F)

n = number of inlet streams.

The heat transfer with the surroundings is determined from the mean convective coefficient, mean radiative coefficient, mean conductive coefficient, the fluid-to-wall heat transfer effectivity, and the structural conductances using the generic thermal network described in section 8.3.

10.33.4 Input Parameters

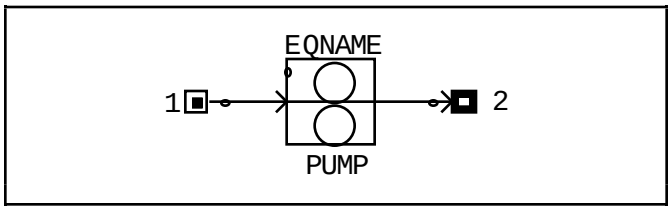
The user input parameters for the PREWAST routine are shown in table 58.

Table 58. PREWAST edit screen.

```
CASE NAME: DEFAULT                                  PREWAST EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                     RECORD NUMBER: 000001
 ____________________________________________________________________________________________________________________________
|                      ** GENERAL INPUT DATA **                      |            ** THERMAL CHARACTERISTICS DATA **                |
|  10  FLUSH WATER RATIO          (LBM/LBM WASTE)    1.0000          | 13 SHELL-ENVIRNMNT CONVECTIVE CONDCTANCE (BTU/HR/F)     0.000 |
|  11  TREATMENT RATIO            (LBM/LBM WASTE)    1.0000          | 14 SHELL-ENVIRNMNT RADIATION F*A*E           (FT^2)     0.000 |
|  12  OUTLET STREAM PRESSURE              (PSIA)   14.7000          | 15 SHELL-ENVIRNMNT CONDUCTIVE CONDCTANCE (BTU/HR/F)     0.000 |
|  23  **  OPEN INPUT LOCATION  **                   0.0000          | 16 MASS-SHELL LINEAR CONDUCTANCE         (BTU/HR/F)     1.000 |
|  24  **  OPEN INPUT LOCATION  **                   0.0000          | 17 WALL-MASS  LINEAR CONDUCTANCE         (BTU/HR/F)     1.000 |
|                                                                    | 18 FLUID-WALL HEAT TRANSFER EFFECTIVITY   (0< X <1) 0.9500000 |
|                    ** RESOURCE TRACKING **                         | 19 THERMAL CAPACITANCE                      (BTU/F)     0.000 |
|  7   POWER                              (WATTS)    0.0000          | 20 INITIAL MASS TEMPERATURE                     (F)    75.000 |
|  8   WEIGHT                               (LBM)    0.0000          | 21 THERMAL SOLUTION RELAXATION CRITERIA   (0< X <1) 0.0000100 |
|  9   VOLUME                              (FT^3)    0.0000          | 22 THERMAL SOLUTION MAX ITERATION COUNT               100.000 |
|____________________________________________________________________|____________________________________________________________|
|                       ** BENCHMARK DATA **                         |                    ** OUTPUT DATA **                       |
|                             MIN          NOM          MAX          | 25 PRESENT OUTLET PRESSURE              (PSIA)     0.000      |
|   WASTE FLOW    (LBM/HR)    0.000        0.000        0.000        | 26 PRESENT MIXTURE FLOW RATE          (LBM/HR)     0.000      |
|   FLUSH FLOW    (LBM/HR)    0.000        0.000        0.000        | 27 PRESENT MASS TEMPERATURE                (F)     0.000      |
|   TREAT FLOW    (LBM/HR)    0.000        0.000        0.000        | 28 PRESENT SHELL TEMPERATURE               (F)     0.000      |
|   MIXT  FLOW    (LBM/HR)    0.000        0.000        0.000        | 29   **  OPEN OUTPUT LOCATION  **                  0.000      |
|____________________________________________________________________|____________________________________________________________|
```

10.34 PUMP—PUMP Simulation Description

The PUMP component simulates the flow driving and pressure rise characteristics of centrifugal and positive displacement pumps or fans. The component only allows mass flow from the inlet stream 1 to the outlet stream 2 (no backflow allowed for inactive pump). It is an "active" device with respect to mass flow—it draws flow from the upstream component and feeds it to the downstream component.



10.34.1 General Assumptions

The PUMP routine is based on the assumptions that the fluid is incompressible and that no fluid leakage occurs. All flow calculations are made on a mass rate basis.

10.34.2 Mass And Pressure Calculations

The PUMP component provides the primary means of driving the flows for the whole CASE/A solution system. It drives the mass flow by drawing mass from the upstream component and feeding it to the downstream component. It may be used for open-cycle systems that draw mass from mass accumulation components such as the STORE or CABIN as well as for closed-cycle systems that do not contain accumulation components. The open-cycle mode simply draws fluid from the upstream component and the fluid composition is determined by that upstream component. The closed-cycle mode allows the user to set the fluid composition to that of any one of the simulation constituents, with the restrictions that the fluid will be composed entirely of that constituent and that there are no flow splits that result in an outgoing mass from the loop. The user may set the cycle mode of the PUMP by setting the fluid code in the input section of the data base to the appropriate value:

A. A value of 0.0 indicates the open-cycle mode.

B. A value greater than 1.0 activates the closed-cycle mode and indicates the constituent number of the fluid (2 to 50).

The default is the open-cycle mode. The calculations for the two modes are identical except that the closed-cycle mode sets the fluid composition instead of accepting the upstream fluid as in the open-cycle mode.

The component offers four modes of operation for the pressure rise versus mass flow rate characteristics of the PUMP. The PUMP may be deactivated by setting the input mass flow rate to 0.0 for all four modes of operation. The PUMP inlet pressure is set by the upstream component for all modes unless the inlet stream accumulator option for the PUMP is activated. The PUMP sets the mass flow rate and outlet pressure depending on the operation mode.

The accumulator option allows the user to set the PUMP inlet pressure to a constant value that acts as a boundary pressure to the upstream component. This option is activated by setting the inlet pressure in the input section of the data base to a nonzero value. The default value of 0.0 indicates that the inlet pressure will be set by the upstream component. The user should be aware that, if care is not taken in the use of this option, an unrealistic case at the inlet to the PUMP may easily be created that will result in nonconvergence of the system pressure solution. The four modes of operation of the PUMP are described below.

10.34.3 Specified Mass Flow Rate With Specified Outlet Pressure

This option simply sets the mass flow rate and the pump outlet pressure to the values specified as input parameters in the data base. It is activated by setting both the mass flow rate and outlet pressure to positive nonzero values. There is no attempt to match the downstream network pressure or flow characteristics. Therefore, this option is usually most applicable to systems that have no downstream boundary pressures or that require an upstream boundary pressure.

10.34.4 Specified Mass Flow Rate With Specified Pressure Rise

This option sets the mass flow rate to the value specified as an input in the data base and sets the outlet pressure to the sum of the inlet pressure and the specified pressure rise across the pump. It is activated by setting the mass flow rate and the first constant of the pump characteristic curve equation to nonzero values. As in the first mode, there is no attempt to match the downstream network pressure or flow characteristics.

10.34.5 Specified Mass Flow Rate With Variable Pressure Rise

This option simulates the action of an ideal positive displacement pump that provides a constant flow rate but matches the downstream pressure losses. The PUMP routine relies on the pressure feedback functions of the system pressure solution routines to set the outlet pressure correctly. The routine does not set the outlet pressure itself, since it is independent of the inlet pressure but dependent on the mass flow rate. The system pressure solution routines are described in section 8.2. This option is activated by setting the mass flow rate to a nonzero value and the outlet pressure of the PUMP to a value of 0.0. This is the default option.

10.34.6 Variable Mass Flow Rate With Variable Pressure Rise

This option simulates the action of a centrifugal pump whose flow rate is dictated by the downstream pressure losses and the characteristic curve of the PUMP. The solution algorithm for this option is based on an implicit iteration on the mass flow rate versus the pressure rise across the PUMP to achieve balancing of the pump with the system. As in the third mode, the PUMP routine relies on the feedback functions of the system pressure solution routines to set the outlet pressure to the correct value for a given mass flow rate. The PUMP routine utilizes the current system iteration pass values of the inlet and outlet pressures to reset the mass flow rate. This is accomplished by finding the root of the fourth degree polynomial curve for the pressure drop specified as a function of flow rate:

$$\Delta P = A_o + A_1\dot{m}_b + A_2\dot{m}_b^2 + A_3\dot{m}_b^3 + A_4\dot{m}_b^4 ,$$

where,

ΔP = the pressure rise across the PUMP

$\dot{m}_b$ = the balanced mass flow rate of the PUMP (lbm/h)

A_x = the characteristic curve coefficients.

The equation is solved for the flow rate using the Newton-Raphson Method of solution for polynomials. Then this value is adjusted by a damping factor, specified by the user as an input parameter, to arrive at the next iteration value of the flow rate for the PUMP:

$$\dot{m}_n = \dot{m}_o + K(\dot{m}_b - \dot{m}_o) ,$$

where,

$\dot{m}_n$ = the next iteration mass flow rate of the PUMP (lbm/h)

$\dot{m}_o$ = the last iteration mass flow rate of the PUMP (lbm/h)

K = the centrifugal flow damping factor.

This procedure is repeated each time the PUMP routine is executed until the system pressure and flow convergence criteria are met for the whole system flow network.

10.34.7 Thermal Calculations

An R-C network analysis technique is utilized in the thermal calculations and is based on the generic network discussed in section 8.3 that allows for three modes of heat transfer with the environment: convection, conduction, and radiation. The total mass of the PUMP package is lumped together to form the mass diffusion node. The network solution calculates the fluid outlet and structural temperatures based on the user-specified conductances, thermal capacitance of the mass node (motor), and environment and fluid inlet boundary temperatures. There are four linear conductances the user must provide as inputs along with the emissive area term for radiation and a fluid-to-wall heat transfer effectivity. Also, the user must specify an initial mass temperature, a relaxation criterion for all temperatures, and a maximum iteration count not to be exceeded in the solution routine. The solution routine also allows for heat loads to the mass and external

shell nodes. In the PUMP calculations, the shell node heat load is set to zero and the mass node heat load is set to the amount of power dissipated by the pump electrical motor as heat. This value is calculated according to:

$$\dot{Q}_d = \dot{W}_m - \dot{W}_i \ ,$$

where,

$\dot{Q}_d$ = the power dissipated as heat by the pump motor (Btu/h)

$\dot{W}_m$ = the total power required by the pump motor (Btu/h)

$\dot{W}_i$ = the actual power required to drive the impeller (Btu/h).

The total power required by the pump motor is given by:

$$\dot{W}_m = \frac{\dot{W}_i}{\eta_m} \ ,$$

where,

η_m = the electrical efficiency of the pump motor ($0 <= \eta_m <= 1.0$).

The actual power required to drive the pump impeller is given by:

$$\dot{W}_i = \frac{\dot{W}_h}{\eta_i} \ ,$$

where,

$\dot{W}_h$ = the hydraulic power required to pressurize the fluid (Btu/h)

η_i = the isentropic efficiency of the pump impeller ($0 <= \eta_i <= 1.0$).

The hydraulic power required to pressurize the fluid is given by:

$$\dot{W}_h = \frac{\dot{m}\ 144\ \Delta P}{778\ \rho} \ ,$$

where,

$\dot{m}$ = the fluid mass flow rate (lbm/h)

ΔP = the pressure rise across the pump (lb/in^2 absolute)

ρ = the fluid density (lbm/ft^3).

The heat load applied to the fluid due to the impeller inefficiency is given by:

$$\dot{Q}_f = \dot{W}_i - \dot{W}_h \ ,$$

where,

$\dot{Q}_f$ = the heat load applied to the fluid by the impeller (Btu/h).

An option has been included to allow the user to override the calculated motor heat load by setting the value to a specified input value. The motor heat load is reset to this value in the calculation of the motor mass temperature, but the fluid heat load from the impeller inefficiency remains at the internally calculated value. Therefore, the user should set the motor and impeller efficiencies to 1.0 if the user desires only the specified heat load on the motor to be dissipated to the fluid. This option is activated by setting the motor heat load flag to 1.0 and the motor heat load to the desired value. The default settings disable this option with values of 0.0 for the flag and the heat load.

10.34.8 Input Parameters

The PUMP data base parameters and their default settings are shown in table 59. The default operation mode of the PUMP is a deactivated positive displacement type with motor and impeller efficiencies of 70 percent. The thermal conductance values are arbitrary settings, and the thermal capacitance is set to 0.0 to provide steady-state operation. The default fluid code is 0.0 to configure the PUMP to the open-cycle mode.

Reference

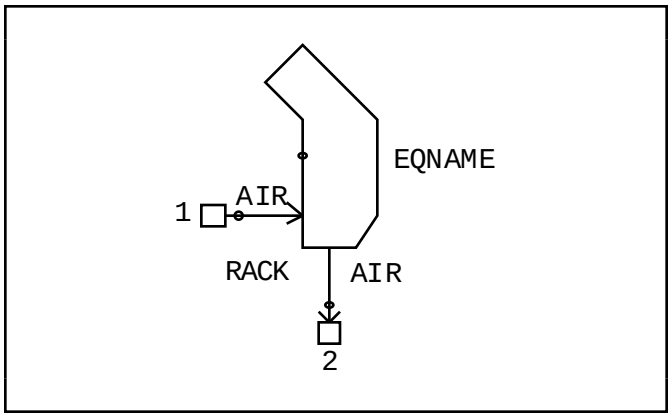
1. Reynolds, W.C., and Henry, C.: “Engineering Thermodynamics.” McGraw-Hill Book Company, 1977.

Table 59. PUMP edit screen.

```
CASE NAME: DEFAULT                                   PUMP EDIT SCREEN                                   LAST UPDATED: 890215
SUBSYSTEM:              COMPONENT:                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|            ** GENERAL INPUT DATA **            |   ** THERMAL CHARACTERISTICS DATA **    |  ** CENTRIFUGAL PUMP OPTION CONSTANTS ** |
| 10 MASS FLOWRATE          (LBM/HR)     0.000 | 22 SHEL-ENV CONVCT (BTU/H/F)     0.200 | 35 FLOW DAMPING FACT (0<X<1)   0.10000000 |
| 11 INLET PRESSURE           (PSIA)     0.000 | 23 SHEL-ENV RADIAT FAE (FT2)     0.800 | 13 CHARCTRISTIC CURVE CON A0  0.00000E+00 |
| 12 OUTLET PRESSURE          (PSIA)     0.000 | 24 SHEL-ENV CONDCT (BTU/H/F)     1.000 | 14 CHARCTRISTIC CURVE CON A1  0.00000E+00 |
| 18 IMPELLER EFFICIENCY    (0<X<1)      0.700 | 25 MASS-SHEL COND  (BTU/H/F)     1.000 | 15 CHARCTRISTIC CURVE CON A2  0.00000E+00 |
| 19 MOTOR EFFICIENCY       (0<X<1)      0.700 | 26 WALL-MASS COND  (BTU/H/F)    50.000 | 16 CHARCTRISTIC CURVE CON A3  0.00000E+00 |
| 20 SPECIFIED MOTR QFLAG (0=N,1=Y)      0.000 | 27 FLUID-WALL EFFECT (0<X<1)     0.900 | 17 CHARCTRISTIC CURVE CON A4  0.00000E+00 |
| 21 SPECIFIED MOTOR QLOAD (BTU/HR)      0.000 | 28 THERMAL CAPACTNCE (BTU/F)     0.000 | 36 ** OPEN INPUT LOCATION **        0.000 |
| 34 FLUID CODE(0=?, 2-50=CNSTIT #)      0.000 | 29 INITIAL MASS TEMP     (F)    75.000 | 7  POWER                (WATTS)      0.000 |
| 32 ** OPEN INPUT LOCATION **           0.000 | 30 THERMAL RELAX CRITERIA    0.0000100 | 8  WEIGHT                 (LBM)      0.000 |
| 33 ** OPEN INPUT LOCATION **           0.000 | 31 THERMAL SOLUTN MAX COUNT    100.000 | 9  VOLUME                (FT^3)      0.000 |
|_____________________________________________|_________________________________________|_________________________________________|
|                        ** BENCHMARK DATA **                          |                    ** OUTPUT DATA **                    |
|                                MIN         NOM         MAX           | 37  MAX CAPACITY FLOW RATE       (LBM/HR)     0.000          |
|  PUMP INLET PRESS  (PSIA)     0.0000      0.0000      0.0000        | 38  PRESENT PRESSURE RISE         (PSID)     0.000          |
|  PUMP OUTLET PRESS (PSIA)     0.0000      0.0000      0.0000        | 39  PRESENT FLOW CONVERGENCE FLAG            0.000          |
|  PUMP PRESS RISE   (PSIA)     0.0000      0.0000      0.0000        | 40  PRESENT MASS TEMPERATURE          (F)     0.000          |
|  PUMP FLOW RATE  (LBM/HR)     0.0000      0.0000      0.0000        | 41  PRESENT SHELL TEMPERATURE         (F)     0.000          |
|  PUMP POWER       (WATTS)     0.0000      0.0000      0.0000        | 42  ** OPEN OUTPUT LOCATION **                0.000          |
|_____________________________________________________________________|_____________________________________________________________|
```

10.35 RACK—RACK Simulation Description

The RACK routine provides for the thermal simulation of an electronic component configured inside a rack cooled primarily by forced convective mechanisms. The nodal network used in modeling the rack is shown in figure 67. It consists of four arithmetic nodes (base plate, shell, fluid outlet, and rack), four boundary nodes (fluid inlet and cabin nodes), and one diffusion node. A mass-shell approach is used to model the electronic component. The mass node is allowed to reject heat to the base plate, shell, and rack structure. A simple delta network is used for the simulation of the base plate to fluid stream heat transfer process. Delta effectivities are calculated internally to the routine as a function of the flow capacity rate and a user specified convective film conductance (hA).



10.35.1 General Discussion

The component shell node is coupled radiatively to the mass and rack nodes. The rack is allowed to convect, radiate, and conduct to the cabin nodes (i.e., bulk air, structure, and MRE, respectively).

10.35.2 Thermal Calculations

The nodal equations are solved simultaneously by a block iteration method in which the diffusion nodal equation is expressed in a backwards finite differenced implicit format. A check on numerical integration stability is provided via nodal time constant considerations. If necessary, the rack routine will reduce the integration time step and loop internally until system and component time are compatible.

10.35.3 Pressure Calculations

Pressure drop through the rack is based on user specified equivalent line length and equivalent line diameter. A more detailed discussion is given in section 8.2.5.

10.35.4 Input Parameters

The input parameters for the RACK routine are shown in table 60.

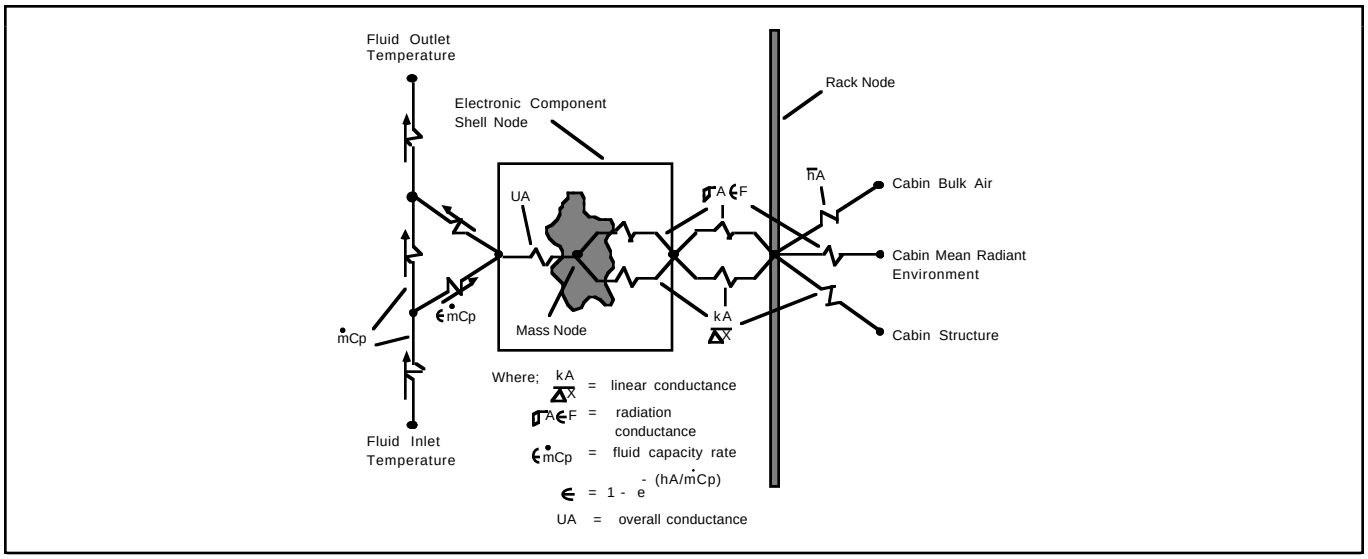


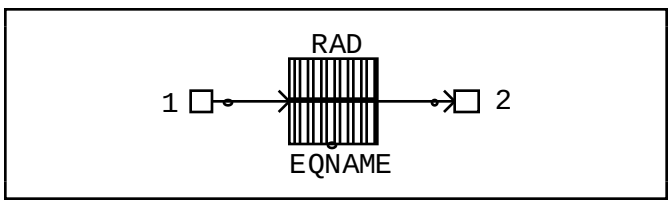
Figure 67. RACK thermal network.

Table 60. RACK edit screen.

```
CASE NAME: DEFAULT            RACK EDIT SCREEN            LAST UPDATED: 890215
SUBSYSTEM:                   COMPONENT:                 RECORD NUMBER: 000001
_____________________________________________________________________________
|        ** GENERAL INPUT DATA **          |                                |
|10 THERMAL LOAD          (WATTS)  500.00 |      ** RESOURCE TRACKING **   |
|11 INITIAL MASS TEMPERATURE  (F)   70.00 |7 POWER            (WATTS)    0.00 |
|12 THERMAL CAPACITANCE   (BTU/F)   25.00 |8 WEIGHT             (LBM)    0.00 |
|                                          |9 VOLUME            (FT^3)   0.000 |
|      ** THERMAL INTERFACE DATA **        |                                |
|15 FLUID-WALL G    (BTU/HR/F)     460.00 |   ** PRESSURE DROP PARAMETERS ** |
|16 MASS-WALL G     (BTU/HR/F)      35.00 |13 EQUIVALENT LENGTH  (FT)    1.00 |
|17 MASS-SHELL G  (BTU/HR/R^4)  0.171E-10 |14 EQUIVALENT DIA     (IN)   6.000 |
|18 MASS-RACK G     (BTU/HR/F)      1.100 |                                |
|19 SHELL-RACK G  (BTU/HR/R^4)  0.171E-10 |                                |
|20 RACK-BULK AIR G (BTU/HR/F)      2.600 |                                |
|21 RACK-STUCT G    (BTU/HR/F)      3.400 |                                |
|22 RACK-MRE G    (BTU/HR/R^4)  0.171E-09 |                                |
|_________________________________________|________________________________|
|               ** BENCHMARK DATA **                |     ** OUTPUT DATA **  |
|                     MIN        NOM        MAX    |                        |
| MASS TEMP    (F)  0.00E+00   0.00E+00   0.00E+00|23 RCK FACE TMP (F)    0.00 |
| AIR OUT TEMP (F)  0.00E+00   0.00E+00   0.00E+00|24 MASS TEMP    (F)    0.00 |
|____________________________________________________|________________________|
```

10.36 RAD—Radiator Simulation Description

Heat rejection from an active thermal control subsystem to space is provided through the radiator (RAD) component. This routine simulates performance of a single-phase space radiator element. Multiple elements can be attached in series or parallel to form an entire space radiator system. Heat rejection characteristics of the radiator are based on user inputs for radiator geometry, radiator optical properties, and radiative sink temperature. Heat transfer between the working fluid and the radiator is determined through empirical relations involving the Reynolds, Prandtl, and Nusselt numbers. Thermal capacitance of the radiator is accounted for in the solution of the radiator thermal network. A user-input radiator fin efficiency is allowed by this component to account for nonidealities such as temperature gradients within the radiator fin. The RAD outlet temperature is calculated with a delta network scheme. The RAD does not generate flow; therefore, an active component must be upstream of the inlet. Pressure drop through the RAD is determined by a user-supplied equivalent line length and equivalent line diameter. The RAD data base parameters are shown in table 61.



Discussion

The RAD utilizes the thermal network described in section 8.3.3, and its adaptation for the RAD is shown schematically in figure 68. The radiator pipe is considered to have no mass (arithmetic node), while the radiator fin itself has a thermal capacitance (isothermal, diffusion node). This thermal network allows heat exchange with the space environment through radiative pathways only. Thermal resistance due to the pipe wall can be accounted for by the routine. Heat transfer from the bulk fluid to radiator wall is dependent upon fluid velocity, density, specific heat viscosity, and thermal conductivity. For a given set of operational conditions, the routine calculates an overall fluid-to-wall convective heat transfer coefficient that is used in the delta network described in section 8.3.2 to calculate the RAD fluid outlet temperature illustrated by the following steps:

A. Calculate Reynolds number:

$$Re = \frac{DV\rho}{\mu} .$$

B. Calculate Prandtl number:

$$Pr = \frac{C_p\mu}{k} .$$

C. Calculate Nusselt number:

$$Nu = f(\mu, Re, Pr) .$$

D. Calculate heat transfer coefficient:

$$h = \frac{NuD}{k} .$$

E. Calculate fluid-to-wall effectivity:

$$\varepsilon = 1 - \exp(hA/\dot{m}C_p) ,$$

where,

D = pipe diameter, ft

V = fluid velocity, ft/s

ρ = fluid density, lb/ft^3

μ = fluid viscosity, lb/ft/h

C_p = fluid specific heat, Btu/lb/°F

k = fluid thermal conductivity, Btu/ft/h/°F

A = heat exchange area, ft^2

$\dot{m}C_p$ = fluid heat capacity rate, Btu/h/°F.

It should be noted that temperature gradients within the radiator fin or a particular fin geometry can be accounted for by proper modification of the radiator fin efficiency data base parameter. Also, the radiative sink temperature can be modified dynamically with the OPS code to simulate various orbital maneuvers.

Pressure drop through the RAD is found by conventional pressure drop calculation methods utilizing a user-supplied equivalent length and equivalent diameter. See section 8.2.5 for a detailed explanation of the pressure drop calculations.

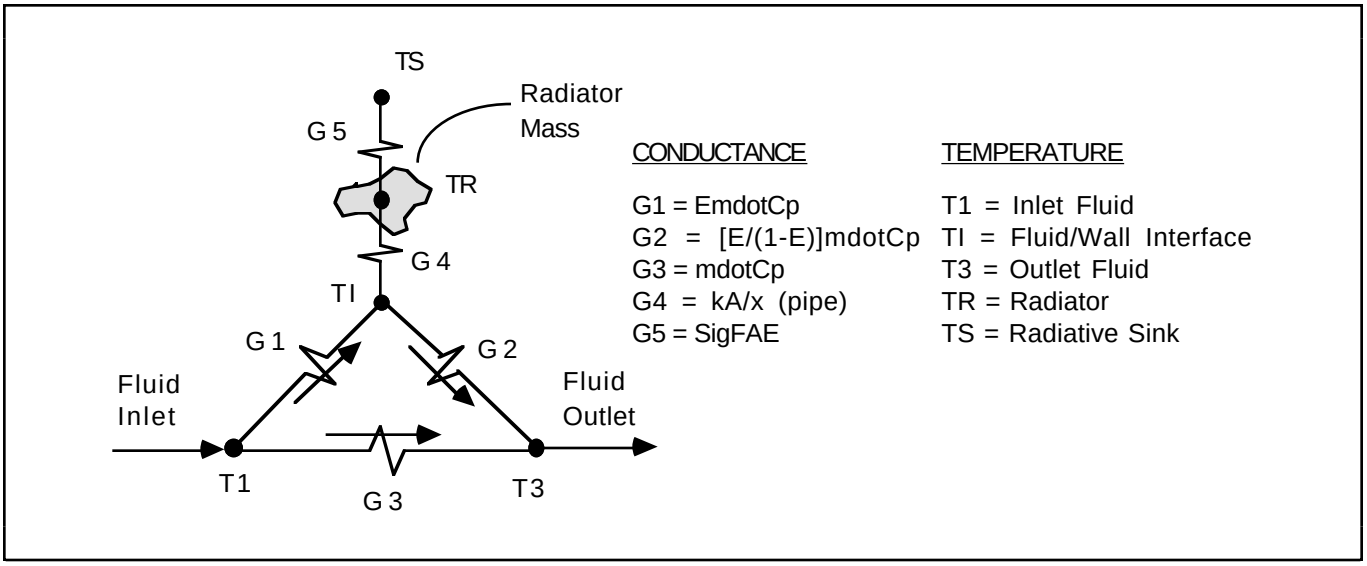


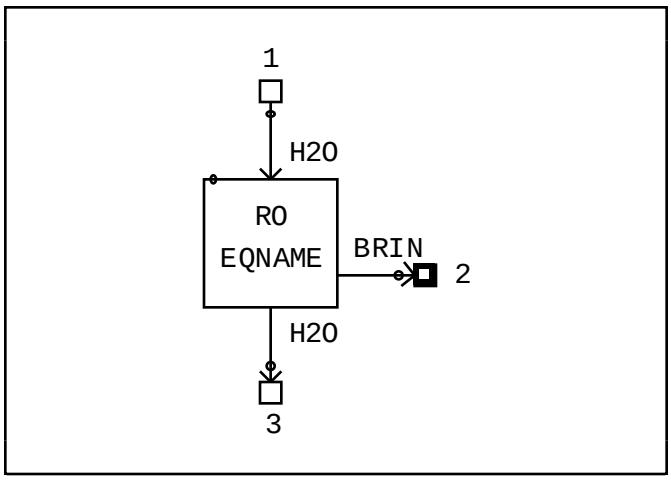
Figure 68. RAD thermal network.

Table 61. RAD edit screen.

```
CASE NAME: DEFAULT              RAD EDIT SCREEN              LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                         RECORD NUMBER: 000001
_________________________________________________________________________________
|          ** GENERAL INPUT DATA **          |     ** THERMAL PARAMETERS **       |
| 13 RADIATOR PIPE LENGTH     (FT)   40.00 | 11 RAD FIN EFFICIENCY     0.80000 |
| 14 RAD PIPE INSIDE DIAMETER (IN)    3.00 | 12 RADIATOR CAP  (BTU/F)  150.000 |
| 15 RAD PIPE WALL THICKNESS  (IN)    0.30 | 10 RADIATOR F-A-E (FT^2)    80.00 |
|                                          | 17 FLUID K (BTU/HR/FT/F)   115.00 |
| 18 INITIAL RAD TEMPERATURE   (F)   45.00 | 16 PIPE K  (BTU/HR/FT/F)     0.05 |
| 20 SINK TEMPERATURE          (F)  -85.00 |      ** RESOURCE TRACKING **       |
|                                          | 7 POWER          (WATTS)     0.00 |
| 21 MAX THERMAL LOOPS                50.0 | 8 WEIGHT           (LBM)     0.00 |
| 22 ENERGY BALANCE ERROR      (%)    0.00 | 9 VOLUME          (FT^3)     0.00 |
|_________________________________________|_________________________________|
|             ** BENCHMARK AND OUTPUT DATA **                                   |
|                      MIN            NOM            MAX                       |
| HEAT TRANSFER (W)  0.00E+00       0.00E+00       0.00E+00                    |
| OUTLET TEMP   (F)  0.00E+00       0.00E+00       0.00E+00                    |
| 24 RADIATOR TEMP      (F)    0.00                                            |
| 25 PIPE WALL TEMP     (F)    0.00                                            |
| 26 FLUID OUTLET TEMP  (F)    0.00                                            |
| 27 HEAT REMOVED   (WATTS)    0.00                                            |
|_________________________________________________________________________________|
```

10.37 RO—Reverse Osmosis Simulation Description

The RO routine simulates the performance of a RO module. The operating conditions, module physical dimensions, solute coefficients, and membrane permeability factors are defined by the user. The constituents are rejected by the membrane according to the user-defined solute coefficients. This routine channels the concentrated contaminant stream to external brine storage or to further processing via stream 2. The product stream is designated as stream 3. The RO routine does not generate flow; therefore, an active component must be upstream of the unit to ensure a proper simulation.



10.37.1 General Discussion

In the simulated RO units, the waste water flows inside porous hollow tubes with the membrane deposited on the inside. The processed water is collected in an outer pressure vessel shell. The spiral-wound configurations may also be modeled by calculating an equivalent tube diameter and length that will approximate the actual flow conditions occurring in the module. The routine is not applicable to the configuration in which the waste water flows into the pressure vessel with the product water passing into the hollow tubes.

10.37.2 Mass And Pressure Calculations

The user provides the membrane-waste system coefficients for a given set of operating conditions and module physical dimensions. From these data, the routine will predict the performance of a module of any physical dimensions and operating conditions. The governing equations were obtained from references 1 and 2.

The user provides the product water pressure, three module physical dimensions, three coefficients for each solute to be tracked, two membrane constants, and 10 heat transfer parameters. Each input is discussed in detail in the following paragraphs.

The product water pressure is the pressure maintained on the shell side of the module for hollow-tube configurations and the pressure maintained in the central collection tube for the spiral-wound configuration. The user may toggle the pressure feedback function of the solution system by setting the water pressure value to zero. This option causes the downstream component to dictate the product pressure.

The three physical dimensions are the number of tubes, the inside diameter of the tubes, and the length of each tube. For the hollow-tube configuration, these values are easily determined. However, for the spiral-wound configuration, equivalent values must be calculated to approximate the flow conditions. For this configuration, the number of tubes is entered as 1, the length of each tube is set equal to the module length parallel to the direction of waste flow, and the equivalent tube diameter is calculated as follows:

$$D_{eq} = \frac{2LX}{L+X},$$

where L is the length in feet of the unwound membrane perpendicular to the direction of waste flow, and X is the size (in feet) of the space between adjacent membrane surfaces through which the waste flows.

The three coefficients for each solute are the solute diffusivity in the solvent, the osmotic pressure factor of the solute in the solvent, and the intrinsic rejection of the solute by the membrane. For noncompetitive solutes, these coefficients should be evaluated with pure water as the solvent. For competitive solutes, the coefficients should be evaluated with the waste water as the solvent. A trial and error comparison might be necessary to “tune” these coefficients for a particular membrane-waste system. Each

coefficient is discussed in detail below. All diffusivities are evaluated at 75 ˚F and atmospheric pressure. The routine accounts for the variation of diffusivity with temperature by the following equation:

$$D_2 = D_1(T_2/T_1)(\mu_1/\mu_2)\,,$$

1 = reference value,

2 = actual value,

where μ is the viscosity of water. It is assumed that the diffusivity is independent of pressure since liquid water is very nearly incompressible.

The osmotic pressure factor is given by the following equation evaluated at any conditions:

$$K = \pi/[(R)(T)(C)]\,,$$

where,

π = osmotic pressure (lbf/ft^2)

R = universal gas constant = 1,545.33 (ft-lbf/lbmole-R)

T = absolute temperature (R)

C = solute concentration (lbm/ft^3).

If there is insufficient information available for the equation above, the following may be used:

$$K = \frac{n}{M}\,,$$

where,

n = number of cations and anions produced by complete dissociation of one "molecule" of solute (e.g., n = 2 for NaCl, n = 1 for Na+, etc.)

M = molecular weight of solute "molecule."

This second equation applies to an ideal solution. A trial and error calculation may be necessary to derive the correct values for a given membrane-waste system from experimental data.

The membrane intrinsic rejection for each solute is calculated from rejection data for highly mixed flow (low polarization conditions) by the following equation:

$$R' = \frac{C_w - C_p}{C_w}\,,$$

where,

C_w = solute concentration in waste inlet stream

C_p = solute concentration in product outlet stream.

The two membrane constants are the hydraulic permeability constant and the amount of waste flow per membrane changeout. The membrane hydraulic permeability constant is evaluated for the flow of pure water through the membrane by the following equation:

$$Lp = \frac{Q}{\Delta PA}\,,$$

where,

Q = volumetric flow of water through membrane (ft^3/h)

ΔP = pressure drop across membrane (lbf/ft^2)

A = membrane area (ft^2).

The waste flow per membrane changeout is simply the flow at which the membrane performance begins to decline. At this point, the routine will update the number of membrane modules required during the simulation.

10.37.3 Thermal Calculations

The 10 heat transfer parameters are the convective, radiative, and structural conduction coefficients for heat transfer; a coefficient to determine the heat transfer between the flowing fluid and the internal flow path surfaces; and the thermal solution control constants. The generic thermal solution discussed in section 8.3.3 is utilized.

10.37.4 Input Parameters

The user input parameters discussed above for the RO routine are shown in table 62. The constituent data base parameters are shown in table 63.

References

1. Meares, P., editor: “Membrane Separation Processes.” Elsevier Scientific Publishing Company, 1976.
2. Sourirajan, S.: “Reverse Osmosis.” Academic Press, 1970.

Table 62. RO edit screen.

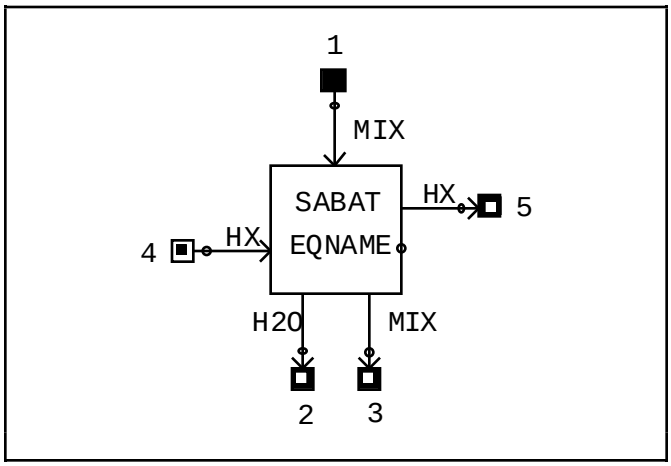
```
CASE NAME: DEFAULT                                          RO EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                       RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|          ** GENERAL INPUT DATA **          |     ** THERMAL CHARACTERISTICS DATA **    |     ** MISCELLANEOUS  PARAMETERS **     |
| 10 MEM HYDRAUL PERM (FT3/LB/HR) 0.100E-05  | 16 SHELL-ENV CONVCT (BTU/HR/F)     0.000  | 26  ** OPEN INPUT LOCATION **     0.000 |
| 11 HYDRAULIC TUBE DIAMETER (FT) 0.820E-03  | 17 SHELL-ENV RAD F*A*E  (FT^2)     0.000  | 27  ** OPEN INPUT LOCATION **     0.000 |
| 12 HYDRAULIC TUBE LENGTH   (FT)     1.000  | 18 SHELL-ENV CONDCT (BTU/HR/F)     0.000  | 175 ** OPEN INPUT LOCATION **     0.000 |
| 13 TOTAL NUMBER OF TUBES         2000.000  | 19 MASS-SHEL CONDCT (BTU/HR/F)     1.000  |                                         |
| 14 PROD OUTLET PRESSURE  (PSIA)    25.000  | 20 WALL-MASS CONDCT (BTU/HR/F)     1.000  |          ** RESOURCE TRACKING **        |
| 15 LBM WASTE FLOW/MEM CHANGOUT  10000.000  | 21 FLUID-WALL EFFECT   (0<X<1) 0.9500000  | 7  POWER              (WATTS)    0.0000 |
|                                            | 22 THERMAL CAPACITNCE  (BTU/F)     0.000  | 8  WEIGHT               (LBM)    0.0000 |
|                                            | 23 INITIAL MASS TEMPRTURE  (F)    75.000  | 9  VOLUME              (FT^3)    0.0000 |
|                                            | 24 THERMAL RELAXATION CRITERIA 0.0000100  |                                         |
| -- CONSTITUENT EMP DATA  (^L  TO OPEN ) _  | 25 THERMAL MAX ITERATION COUNT   100.000  |                                         |
|____________________________________________|___________________________________________|_________________________________________|
|                        ** BENCHMARK DATA **                          |  176  PRESENT UNIT’S SERVICE TIME         (HR)     0.000  |
|                                MIN         NOM         MAX           |  177  NUMBER OF UNITS REQUIRED                     0.000  |
| BRINE  PRESS DROP  (PSID)       0.000       0.000       0.000        |  178  PRESENT BRINE PRESSURE DROP       (PSID)     0.000  |
| WASTE  INLET FLOW (LBM/H)       0.000       0.000       0.000        |  179  PRESENT WASTE IN FLOW RATE      (LBM/HR)     0.000  |
| BRINE  OUTLT FLOW (LBM/H)       0.000       0.000       0.000        |  180  PRESENT MASS TEMPERATURE            (F)     0.000  |
| PRDUCT OUTLT FLOW (LBM/H)       0.000       0.000       0.000        |  181  PRESENT SHELL TEMPERATURE           (F)     0.000  |
|                                                                      |  182  **  OPEN OUTPUT LOCATION  **                0.000  |
|______________________________________________________________________|__________________________________________________________|
```

Table 63. RO constituents edit screen.

```
CASE NAME:  DEFAULT                                     RO CONSTITUENT SCREEN                           LAST UPDATED: 890215
SUBSYSTEM:  DEFAULT       COMPONENT:                                                                   RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|   LABEL    | INTR REJ | DIFFUSIV | OSMOT K |  LABEL    | INTR REJ | DIFFUSIV | OSMOT K |  LABEL    | INTR REJ | DIFFUSIV | OSMOT K  |
|            | 0<=X<=1  |  FT2/HR  |LBMOLE/LB|           | 0<=X<=1  |  FT2/HR  |LBMOLE/LB|           | 0<=X<=1  |  FTR2/HR | LBMOLE/LB|
|____________|__________|__________|_________|___________|__________|__________|_________|___________|__________|__________|_________|
|CONSTIT 2   0.9000000 0.100E-04 0.100E-03 CONSTIT 19  0.9000000 0.100E-04 0.100E-03 CONSTIT 36  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 3   0.9000000 0.100E-04 0.100E-03 CONSTIT 20  0.9000000 0.100E-04 0.100E-03 CONSTIT 37  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 4   0.9000000 0.100E-04 0.100E-03 CONSTIT 21  0.9000000 0.100E-04 0.100E-03 CONSTIT 38  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 5   0.9000000 0.100E-04 0.100E-03 CONSTIT 22  0.9000000 0.100E-04 0.100E-03 CONSTIT 39  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 6   0.9000000 0.100E-04 0.100E-03 CONSTIT 23  0.9000000 0.100E-04 0.100E-03 CONSTIT 40  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 7   0.9000000 0.100E-04 0.100E-03 CONSTIT 24  0.9000000 0.100E-04 0.100E-03 CONSTIT 41  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 8   0.0000000 0.100E-04 0.100E-03 CONSTIT 25  0.9000000 0.100E-04 0.100E-03 CONSTIT 42  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 9   0.9000000 0.100E-04 0.100E-03 CONSTIT 26  0.9000000 0.100E-04 0.100E-03 CONSTIT 43  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 10  0.9000000 0.100E-04 0.100E-03 CONSTIT 27  0.9000000 0.100E-04 0.100E-03 CONSTIT 44  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 11  0.9000000 0.100E-04 0.100E-03 CONSTIT 28  0.9000000 0.100E-04 0.100E-03 CONSTIT 45  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 12  0.9000000 0.100E-04 0.100E-03 CONSTIT 29  0.9000000 0.100E-04 0.100E-03 CONSTIT 46  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 13  0.9000000 0.100E-04 0.100E-03 CONSTIT 30  0.9000000 0.100E-04 0.100E-03 CONSTIT 47  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 14  0.9000000 0.100E-04 0.100E-03 CONSTIT 31  0.9000000 0.100E-04 0.100E-03 CONSTIT 48  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 15  0.9000000 0.100E-04 0.100E-03 CONSTIT 32  0.9000000 0.100E-04 0.100E-03 CONSTIT 49  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 16  0.9000000 0.100E-04 0.100E-03 CONSTIT 33  0.9000000 0.100E-04 0.100E-03 CONSTIT 50  0.9000000 0.100E-04  0.100E-03 |
|CONSTIT 17  0.9000000 0.100E-04 0.100E-03 CONSTIT 34  0.9000000 0.100E-04 0.100E-03                                            |
|CONSTIT 18  0.9000000 0.100E-04 0.100E-03 CONSTIT 35  0.9000000 0.100E-04 0.100E-03                                            |
|____________________________________________________________________________________________________________________________|
```

10.38 SABAT—SABAT Simulation Description

This routine simulates the SABATIER carbon dioxide reduction process. The mass conversion performance is determined by a chemical equilibrium approach wherein the free stream compositions are calculated from the simultaneous numerical solution of the equilibrium equation and reactor energy balance. The inlet process gas enters the reactor via stream 1. The condensed water vapor exits the SABAT via stream 2. Noncondensables, reaction products, and chemical inerts leave via stream 3. The coolant fluid is pulled into the heat exchangers via stream 4 and exits via stream 5. The SABAT icon configuration defining the respective streams is shown below.



10.38.1 General Assumptions

The following assumptions and simplifications were made during the development of the SABAT routine.

A. The only two reactions modeled are the main Sabatier reaction and the formation of water vapor from incoming oxygen and hydrogen.

B. The reaction of O_2 and H_2 is set at a constant user-defined percentage of the incoming O_2 and occurs before the main reaction.

C. The mass conversion performance of the reactor is based on the assumption that the gaseous reaction products attain chemical equilibrium composition at the catalyst surface.

D. The composition of the process gas is assumed to be constant with radial position in the cylindrical catalyst tube.

E. All chemical species not involved in the main reactions are passed through the reactor as inerts.

F. The nitrogen accumulator and purge functions are not modeled.

10.38.2 SABATIER Subsystem Description

The Sabatier CO_2 reduction subsystem consists of a cylindrical reactor tube, air-cooled condenser, porous plate separator, water accumulator, air cooling fan, and associated plumbing hardware and controller. A schematic of the modeled subsystem is shown in figure 69. The carbon dioxide and hydrogen mixture enters the subsystem through a charcoal filter that removes trace contaminants. The mixture then passes to the preheated reactor where it is converted to water vapor and methane. The reactor contains two successive air-cooled heat exchange zones following an insulated entrance portion. The water vapor, methane, and excess reactant (either CO_2 or H_2) then flow to the air-cooled condenser/separator, where the water vapor is condensed, separated from the gas stream, and pumped out. The water is forced out of the water separator by the pressure differential between the reactant pressure and a spring-loaded accumulator that maintains a constant pressure drop across the porous plate separator. Air cooling flow is supplied to the Sabatier reactor and the condenser/separator by a fan at a fixed rate.

10.38.3 Mass Balance Calculations

The reactor configuration modeled is a single-pass axial flow device containing an insulated primary reaction zone followed by two successive air-cooled zones. The mass conversion performance calculations are based on the Sabatier reaction and equilibrium constant equations combined with the thermal characteristics of the subsystem. The primary Sabatier chemical reaction is exothermic and self-sustaining up to a temperature of about 1,100 °F. The reaction takes place at the surface of the Ruthenium on alumina catalyst:

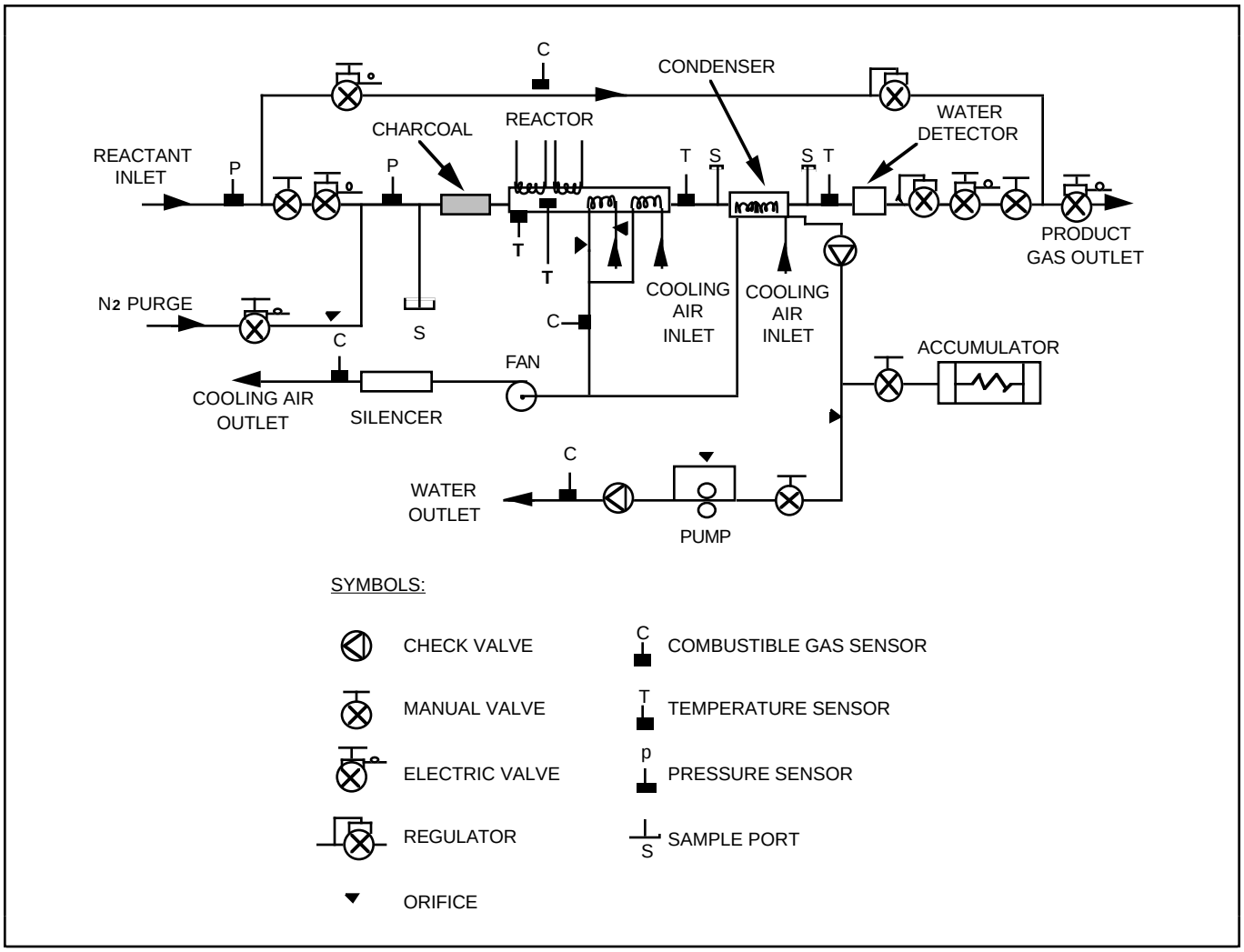


Figure 69. SABAT subsystem schematic.

$$CO_2 + 4\,H_2 \leftrightarrow CH_4 + 2\,H_2O + 43 \text{ kcal.}$$

The reaction rate equation used to calculate the conversion of CO_2 reactant is:

$$R = \frac{A}{T}\exp\left(\frac{-Ea}{RT}\right)\left(PPCO_2^{1/4}\,PPH_2 - \frac{PPCH_4^{1/4}\,PPH_2O^{1/2}}{K_{eq}^{1/4}}\right),$$

where,

R = CO_2 conversion rate, gm moles/sec/cm^3 catalyst

A = 1.112393×10^6 gm moles K/s/cm^3/atm$^{5/4}$

T = catalyst surface temperature, K

$\frac{Ea}{R}$ = 8,553 K

$PPCO_2$ = carbon dioxide partial pressure at catalyst surface, atm

PPH_2 = hydrogen partial pressure at catalyst surface, atm

$PPCH_4$ = methane partial pressure at catalyst surface, atm

PPH_2O = water vapor partial pressure at catalyst surface, atm

Keq = thermodynamic equilibrium constant, atm^{-2}.

The catalyst tube is modeled as six axial nodes through which the reactant-product gas mixture proceeds while undergoing composition and temperature changes. The outlet composition and temperature of each segment becomes the inlet conditions for the next node. The outlet composition at each node is calculated via the rate equation and the primary reaction stoichiometric equation listed above. The reaction rate equation was obtained from reference 1.

It was assumed that thermodynamic equilibrium will be reached at the outlet of each of the six modeled axial nodes along the reactor tube length. The equilibrium constant was calculated according to (see Reference 1):

$$Keq = \exp\left[0.5033\left(\frac{56000}{T^2}\right) + \frac{34633}{T - 16.4 \ln T + 0.00557\,T} + 33.165\right]$$

where,

Keq = thermodynamic equilibrium constant, atm^{-2}

T = catalyst surface temperature, K.

The only secondary reaction modeled is the reaction of gaseous oxygen and hydrogen to form water vapor:

$$O_2 + 2H_2 \leftrightarrow 2H_2O + 57.6 \text{ kcal.}$$

No attempt was made to calculate the equilibrium state of this reaction. Instead, a user-defined reaction efficiency constant is utilized to calculate the rate of conversion of oxygen:

$$R = K\,\dot{N}_{O_2}\,,$$

where,

R = O_2 conversion rate, lbmoles/h

K = reaction efficiency fraction

$\dot{N}_{O_2}$ = O_2 inlet molar flow rate, lbmoles/h.

It is assumed that this reaction is very fast and occurs before the primary Sabatier reaction. Therefore, the composition of the reactant-product gas mixture is first adjusted for this secondary reaction before the main calculations of the Sabatier reaction take place.

A check for the occurrence of condensation at the catalyst surface in each of the four nodes in the two heat HX zones of the reactor tube is performed in addition to the condensation calculations in the condensing HX. The condensation process affects the primary Sabatier reaction via chemical composition alteration as well as by alteration of the heat load applied to the catalyst surface. The change of state of gaseous water vapor to liquid water reduces the water vapor partial pressure, thus affecting the rate of reactants converted to products via the rate equation shown above. The heat load applied to the catalyst surface due to condensation also affects the rate. The condensation check is made by comparing the catalyst surface temperature to the dew point temperature of the gas mixture at the outlet of each node. If the surface temperature is at or below the dew point, the rate of condensation is determined by first calculating the net change in heat load required to maintain the catalyst surface at the dew point. This net heat load change is then divided by the specific heat of vaporization of water to give the rate of condensation. This rate is then used to calculate the change in water vapor partial pressure of the gas mixture at the node outlet:

$$R = \frac{\sum G_{ij}(T_{dew} - Tj) - \sum Qn}{h_{fg}}\,,$$

where,

R = H_2O condensation rate, lbm/h

Gij = thermal heat conductance between nodes i and j, Btu/h/°F

Qn = heat load due to reaction n, Btu/h

hfg = heat of vaporization of H_2O = 1,000 Btu/lbm (assumed constant)

T_{dew} = dew point temperature, °F

Tj = temperature of node j, °F.

10.38.4 Interface Streams

The CASE/A representation of the Sabatier subsystem contains five interfaces or streams through which other components exchange the various fluids involved in the process. These streams are defined according to the “control volume” enclosed by the entire Sabatier subsystem. This control volume for the presently

modeled configuration includes the main reactor unit, the coolant loop (including the coolant fan), the condenser/separator package, and the water accumulator package. The nitrogen purge system is presently excluded from the control volume and no purge functions are modeled.

The inlet reactants stream composition and flow rate are set by the connecting component. The model accepts this flow and copies it into stream 3 (product outlet gas). If the reactor is set to the operation mode via a user-defined constant, the chemical and thermal calculations are performed along the axial length of the reactor. The composition of stream 3 is adjusted along the length until the outlet of the reactor unit is reached. The composition of stream 3 is adjusted as the gas then flows through the condensing HX and a certain amount of water vapor is condensed and separated from the product gas. The final product composition is thus set at the CHX gas outlet.

The composition of stream 2, the condensate outlet, is assumed to be 100-percent liquid water as presently modeled. The periodic cycling of the accumulator is not modeled with the outlet flow rate being set equal to the average condensation rate over the simulation time step. The reactor unit and condenser/separator sections are bypassed if the user flag is set to indicate the inoperative mode. This results in the setting of the product gas composition equal to the inlet gas and the condensate flow equal to zero. The coolant loop is assumed to be continuously operative with the flow rates in each of the three HX sections set by the user. The coolant inlet stream (stream 4) is modeled as an “active puller” stream that pulls flow from the connecting upstream component (usually a CABIN or STORE component). The flow rate is set equal to the sum of all three HX sections while the composition is set by the connecting component. This flow rate and composition are then copied into the coolant outlet stream (stream 5) with no change. The coolant flow is generated continuously when the unit is in the operative or inoperative modes. The user may shut off the coolant loop only by setting the three flow rates equal to zero.

10.38.5 Stream Pressures

The pressure of the inlet reactant’s gas is set by the user via the reactor operating pressure input location. The CASE/A system treats stream 1 as an accumulator device that regulates the pressure via an internal variable volume bladder. The reactor operating pressure is assumed to remain constant along the entire length of the reaction zone as well as through the CHX at this user-specified value.

The product outlet gas (stream 3) and the water outlet (stream 2) pressures are set according to the user-specified values based on one of two methods. If the stream pressures are set to non-zero values in the edit screen, the CASE/A system is instructed to assume that a flow device within the SABAT subsystem regulates their pressures to these values. If the stream pressures are set to 0.0 by the user, the pressures are matched to the downstream flow configuration and/or boundary pressures. This last approach corresponds to the action of a positive displacement flow generation device.

The coolant inlet pressure (stream 4) is set by the upstream connecting components, since it is modeled as the inlet feed to the internal coolant fan. The coolant outlet pressure (stream 5) is set by the user in the same manner as the product outlet gas and the water outlet streams, since the flow is forced out by the fan. All three outlet stream pressure values are defaulted to 0.0 when the SABAT component is located in a case.

10.38.6 Thermal Calculations

The Sabatier subsystem was modeled as two main separate sections: the main catalyst reactor tube unit and the condensing HX/water separator package. The thermal response of the main reactor unit is simulated by the thermal conductor network shown in figure 70. The reactor catalyst bed is divided into six longitudinal sections. Each section has a set of thermal nodes arranged radially that represent the reactant gas, catalyst surface, lumped-mass tube node, insulation or cooling jacket, and the external perforated shield. At each section, the governing mass and energy balance equations are applied, which describe the thermal and chemical response of the reactor structural elements and the free stream reactant gas. There are a total of 45 thermal nodes and 108 conductor paths comprising the thermal network. All nodes are arithmetic except for the six lumped-mass diffusion nodes. There are six boundary nodes representing the reactant inlet gas temperature, the two coolant inlet temperatures, and the three surrounding environment

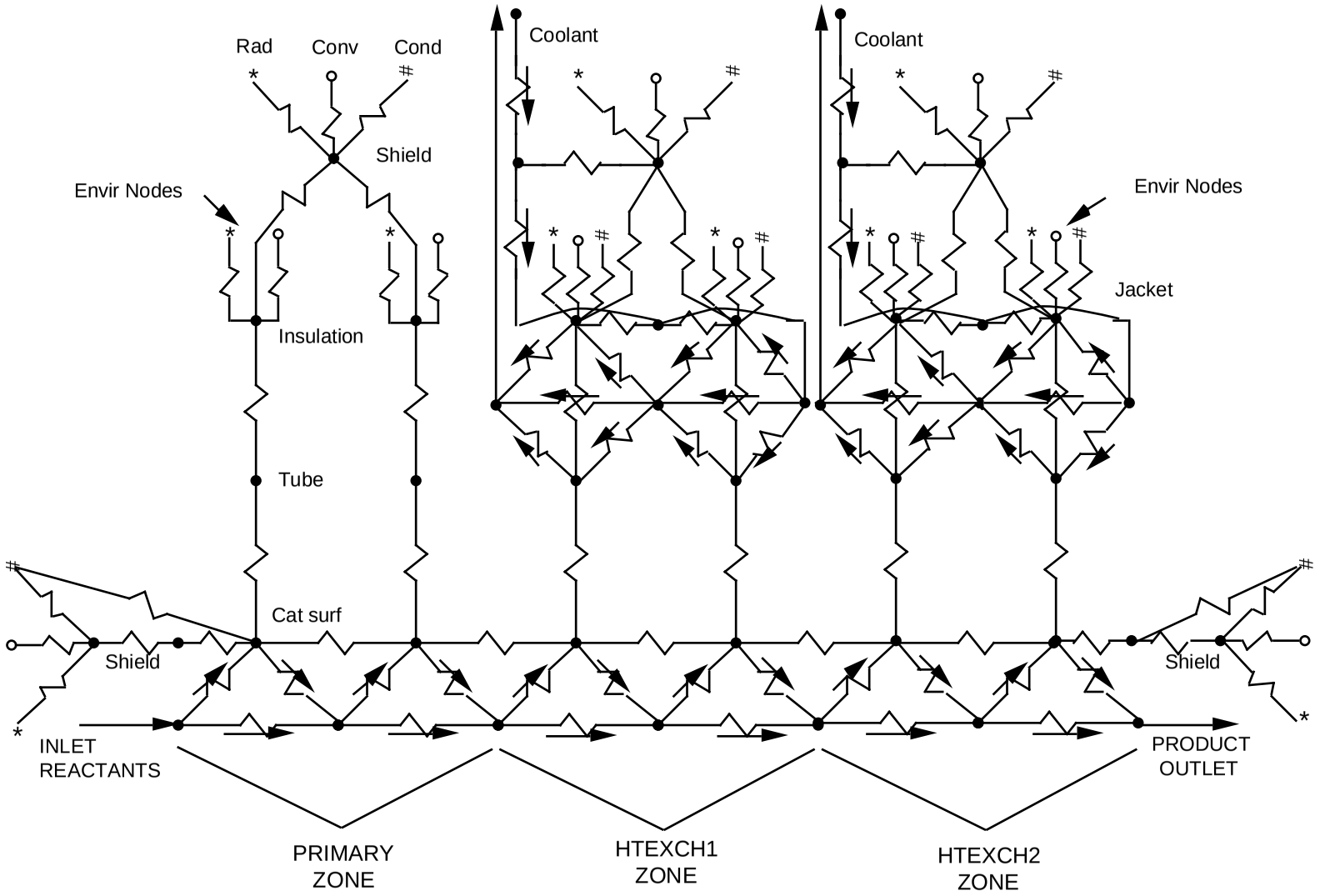


Figure 70. Sabatier reactor thermal network schematic.

temperatures. Four of the thermal nodes represent the shield inlet and outlet endplates, the inlet end insulation, and the outlet spring-loaded endcap of the catalyst tube. All conductors are single-mode energy transfer paths except for those connecting the reactor tube structure to the external shield. These are combined radiation, convection, and conduction energy flow paths.

The delta network is used to simulate the packed bed fluid-to-wall heat transfer from the catalyst surface to the free stream gases. The heat transfer effectivities were set at a constant value of 99.9 percent based on hand calculations.

The heat of reaction is applied to the catalyst surface and is transferred through the radial conductor toward the environment as well as longitudinally to the adjacent section. The heat of reaction was calculated according to (see reference 1):

$$Q = 16.4\mathrm{T} - 0.00557\, T^{2} + 112{,}000\, T^{-1} + 34{,}633\ .$$

where,

Q = heat of reaction, cal/gmmole

T = catalyst surface temperature, K.

The catalyst conductor values are calculated with an effective thermal conductivity for a packed bed given the void fraction, the thermal conductivity of the solid catalyst, and the fluid properties. The equation was obtained from reference 4.

Heat transfer through the insulation surrounding the primary reaction zone is based on the assumption that it is MIN K high-temperature insulation. The temperature variance of the thermal conductivity is recorded in an internal data array.

The thermal response of the coolant loop of the two HX zones is simulated via two coupled delta networks as shown in figure 70. The coupled delta networks simulate the flow of the coolant through the circumferential cooling jackets surrounding the reactor catalyst tube. The delta effectivities were set at 99.9 percent, based on the fact that the serrated fins located inside the passage present a large amount of heat transfer area to the fluid. Also, the fins provide a conduction path between the catalyst tube and the outside jacket. This path is simulated by a radial conductor assuming the material is 347 stainless steel. The thermal conductivity is recorded in an internal data storage array.

Heat transfer from the reactor core insulation is assumed to occur to the exterior shield as well as to the external environment. The transfer to the shield is simulated by a multimode conductor calculated as the sum of the radiation, convection, and conduction terms. The radiation transfer is approximated by concentric cylinder techniques. The conduction transfer is based on a user input mounting area/separation distance value for the support brackets. The thermal conductivity for the conduction term is taken as that of 347 stainless steel. The convection term is simulated based on the gravity field value input by the user as well as by the external forced flow velocity. For the 1-g environment, free convection between the insulation surface and the shield is simulated. For the 0-g environment, the convection term is set equal to the stagnant air gap conductance in the absence of external forced flow. However, for both gravity environments in the presence of forced flow, the highest value of the convection coefficients (forced or free) is taken. The forced convection coefficient is based on cylinders in crossflow. The free convection term is based on a vertically oriented cylinder. Also, convective heat transfer between the insulation and the external environment directly is simulated in the presence of forced flow, assuming the forced flow passes through the perforated shield and passes over the insulation. Finally, radiative heat transfer between the insulation and the external environment through the perforated shield is also simulated. The shield was assumed to be 50-percent open space based on mechanical drawings listing the hole pattern and size.

The heat transfer between the HX jackets to the exterior shield and external environments is modeled in an identical manner, except for the presence of two additional conductor paths. The first is a conductor path to the environment that represents the coolant outlet tube and the mounting bracket, both of which are connected to the jackets. The second additional conductor path is that of convection to the incoming coolant air passing over the jacket before entering the intakes.

The heat losses from the exterior shield consist of radiation to the mean radiant environment, convection (free or forced) to the surrounding bulk air, and conduction through any

external mounting brackets (set by the user). An additional convection path to the incoming coolant air is simulated for the shield nodes surrounding the HX sections. The free convection versus forced convection problem is handled in the same manner as described above. The heat transfer equations were taken from references 2 to 4. The primary reaction zone of the reactor has two electric heaters inserted into the catalyst that are used to bring the catalyst up to self-ignition temperature (about 350 °F) during startup and to ensure that the catalyst does not drop below this temperature during operation. This response is simulated by the simple method of applying the heater thermal load to the catalyst surface nodes in the primary zone when the lumped mass node temperature drops below a user-specified setpoint in the normal operation mode. They are deactivated whenever the catalyst temperature of the first node is above the set point 0 whenever the unit is inoperative. The heater power is set by the user.

The CHX model uses a wet/dry computation algorithm for simulating the thermal performance of the CHX. This technique divides the HX into two separate sections, representing a sensible and a condensing region. The overall UA technique is utilized where the UA value for the CHX is assumed constant and specified by the user. The algorithm assumes film condensation with only water vapor being condensed and no adjustments are made for competing condensables. A detailed description of the algorithm may be found in section 10.11.

10.38.7 Input Parameters

The current SABAT edit screens and default data are shown in tables 64 and 65. The split-screen setup was required due to the large number of inputs made available to the user. Most of the inputs are self-explanatory, since a large number of them are physical dimensions of the reactor unit or various material properties. The input parameters that require special consideration are described below. The spelling of the parameter names correspond to the SABAT edit screen spelling. The numbers on the left correspond to the data base designations used in the CASE/A utility routines GETK and SETK.

10 OPERATION MODE, 0 = OFF 1 = RUN – This flag sets the operating state of the SABAT and can be controlled dynamically from the user OPS logic or by a CONTRLLR component.

72 SHEL-ENV RADATN FAE, Ft2 – This parameter is currently not being used since the CHX external heat losses are not modeled.

73 SHEL-ENV CONV, BTU/HR/F – This parameter is currently not being used since the CHX external heat losses are not modeled.

74 SHEL-ENV COND, BTU/HR/F – This parameter is currently not being used since the CHX external heat losses are not modeled.

Table 64. SABAT default edit screen No. 1.

```
CASE NAME: DEFAULT                          SABAT EDIT SCREEN                                   LAST UPDATED: 890619
SUBSYSTEM:              COMPONENT:                                                            RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________
|        ** GENERAL INPUT DATA **        |                          ** OUTPUT DATA **                                |
|10 OPERATION MODE,    0=OFF    1=RUN    1.|77 CO2 REDUCTION EFFICIENCY, %  0.0000E+00 | 90 HX1 COOLANT OUTLET TEMP, F  0.0000E+00 |
|11 REACT OPERATING PRESS,Psia   17.00   |78 H2  REDUCTION EFFICIENCY, %  0.0000E+00 | 91 HX2 BED SURFACE    TEMP, F  0.0000E+00 |
|12 WATER SEPAR OUT PRESS,Psia  0.0000E+00 |79 CO2 REDUCED (CUMULATIVE),Lb  0.0000E+00 | 92 HX2 BED MASS       TEMP, F  0.0000E+00 |
|13 COOLNT LOOP OUT PRESS,Psia  0.0000E+00 |80 REACTOR BED SURFACE TEMP, F  0.0000E+00 | 93 HX2 COOLING JACKET TEMP, F  0.0000E+00 |
|18 O2 CONVERSION EFFNCY,FRACT  0.0000E+00 |81 REACTOR BED MASS    TEMP, F  0.0000E+00 | 94 HX2 EXTERNAL SHELL TEMP, F  0.0000E+00 |
|28 REACT INITIAL MASS TEMP, F   350.0   |82 REACTOR INSULATION  TEMP, F  0.0000E+00 | 95 HX2 PRODUCT OUTLET TEMP, F  0.0000E+00 |
|29 HTEX1 INITIAL MASS TEMP, F   80.00   |83 REACTOR EXTERN SHEL TEMP, F  0.0000E+00 | 96 HX2 COOLANT OUTLET TEMP, F  0.0000E+00 |
|30 HTEX2 INITIAL MASS TEMP, F   80.00   |84 REACTOR PRODUCT OUT TEMP, F  0.0000E+00 | 97 CONDENS INLT DEWPT TEMP, F  0.0000E+00 |
|31 REACTOR THERMAL CAP, Btu/F  0.1000   |85 HX1 BED SURFACE    TEMP,  F  0.0000E+00 | 98 CONDENSATION RATE,   Lb/hr  0.0000E+00 |
|32 HTEXCH1 THERMAL CAP, Btu/F  0.1000   |86 HX1 BED MASS       TEMP,  F  0.0000E+00 | 99 CONDENS EXT SHELL  TEMP, F  0.0000E+00 |
|33 HTEXCH2 THERMAL CAP, Btu/F  0.1000   |87 HX1 COOLING JACKET TEMP,  F  0.0000E+00 |100 CONDENS PROD OUT   TEMP, F  0.0000E+00 |
|35 CATLYST TUBE INSID DIA, In   1.430   |88 HX1 EXTERNAL SHELL TEMP,  F  0.0000E+00 |101 CONDENS COOLNT OUT TEMP, F  0.0000E+00 |
|36 CATLYST TUBE OUTSD DIA, In   1.500   |89 HX1 PRODUCT OUTLET TEMP,  F  0.0000E+00 |102 AVERAGE HEATER POWER,WATTS  0.0000E+00 |
|37 REACTR CATLYST TUBE LEN,In   4.550   |_____________________________________________|_____________________________________________|
|38 HTEX1  CATLYST TUBE LEN,In   4.500   |                      ** BENCHMARK DATA **                   |          ** RESOURCES **          |
|38 HTEX2  CATLYST TUBE LEN,In   4.500   |                          MIN          NOM          MAX     | POWER,   W  0.0000E+00            |
|14 EXTRNL FORCD FLOW VEL,Ft/s  0.0000E+00 | REACTNT GAS FLOW, Lb/hr  0.0000E+00   0.0000E+00   0.0000E+00| WEIGHT,Lbm  0.0000E+00            |
|34 GRAVITY ENVIRONMENT (0=0g, 1=1g)    0.| PRODUCT GAS FLOW, Lb/hr  0.0000E+00   0.0000E+00   0.0000E+00| VOLUME,Ft3  0.0000E+00            |
|   HEAT TRANSFER DATA (ENTER ^L TO OPEN)_| CONDENSATE  FLOW, Lb/hr  0.0000E+00   0.0000E+00   0.0000E+00|                                   |
|________________________________________|_____________________________________________|_____________________________________________|
```

Table 65. SABAT default edit screen No. 2.

```
CASE NAME: SABAT                                  SABAT THERMAL SCREEN                                          LAST UPDATED: 890619
SUBSYSTEM: DEFAULT       COMPONENT: VALUES                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________________
|              **   REACTOR DATA   **                |                          ** HEAT EXCHANGER DATA **                          |
| 22 HEATER SET POINT TEMPERATURE,  F     350.0      |                                                  HX #1                HX #2     |
| 23 MAXIMUM HEATER POWER,      Watts     200.0      |        COOLANT FLOW IN,          Lb/Hr  15   0.5720         16   0.9540      |
| 24 HEATER WELLS DIAMETER,        In    0.3350      |        COOLANT JACKET THICKNESS,   In  41   0.1600         42   0.1600      |
| 25 INLET END INSUL THICKNESS,    In    0.8000      |        No. OF JACKET RADIAL FINS       43    150.0         44    150.0      |
| 26 INSUL SUPPORT BRACKET A/dx,   In    0.0000E+00  |        JACKT-ENVR MOUNTNG A/dx,    In  46   0.1250E-01     47   0.1250E-01  |
| 40 INSULATION THICKNESS,         In    0.6250      |        EXTERNAL SHELL DIAMETER,    In  49    3.500         50    3.500      |
| 48 EXTERNAL SHELL DIAMETER,      In     3.500      |        EXTERNAL SHELL LENGTH,      In  52    4.700         53    5.500      |
| 51 EXTERNAL SHELL LENGTH,        In     6.800      |        SHELL-ENVR MOUNTNG A/dx,    In  56   0.0000E+00     57   0.0000E+00  |
| 55 SHELL-ENVR MOUNTNG A/dx,      In    0.0000E+00  |        *** OPEN INPUTS ***             68   0.0000E+00     69   0.0000E+00  |
| 27 *** OPEN INPUT ***                  0.0000E+00  |        *** OPEN INPUTS ***             61   0.0000E+00     62   0.0000E+00  |
|_____________________________________________________|_____________________________________________________________________________|
|       ** SOLUTION CONTROL DATA  **        |          **  MISC MATERIAL DATA  **          |          **  CONDENSER DATA  **          |
|19 RELAXATION CRITERION        0.1000E-04  |58 CATALYST PARTICLE DIAMETER, In  0.3940E-01 |17 COOLANT FLOW IN ,  Lb/Hr   99.00      |
|20 SOL GAIN (0<x<1=no dampng)  0.2500      |59 CATALYST VOID FRACTION          0.6800     |71 HX OVERALL UA,   Btu/H/F   20.00      |
|21 MAX No. ITERATIONS           100.0      |60 MOUNTING BRACKET K, Btu/H/Ft/F   10.00     |72 SHELL-ENV RAD FAE,   Ft2  0.0000E+00  |
|63 *** OPEN INPUT ***          0.0000E+00  |65 EXTERNAL SHELL EMISSIVITY       0.1000     |73 SHELL-ENV CONV, Btu/Hr/F  0.0000E+00  |
|64 *** OPEN INPUT ***          0.0000E+00  |66 COOLING JACKET EMISSIVITY       0.4000     |74 SHELL-ENV COND, Btu/Hr/F  0.0000E+00  |
|73 *** OPEN INPUT ***          0.0000E+00  |67 REACTOR INSULATION EMISSIVITY   0.5000     |75 *** OPEN INPUT ***        0.0000E+00  |
|45 *** OPEN INPUT ***          0.0000E+00  |54 *** OPEN INPUT ***              0.0000E+00 |76 *** OPEN INPUT ***        0.0000E+00  |
|___________________________________________|_____________________________________________|_____________________________________________|
```

References

1. Forsythe, R.K., Verostka, C.E., Cusick, R.J., and Blakely, R.L.: “A Study of Sabatier Reactor Operation in Zero ‘G’.” Fourteenth Intersociety Conference on Environmental Systems, 1984, SAE Paper 840936.

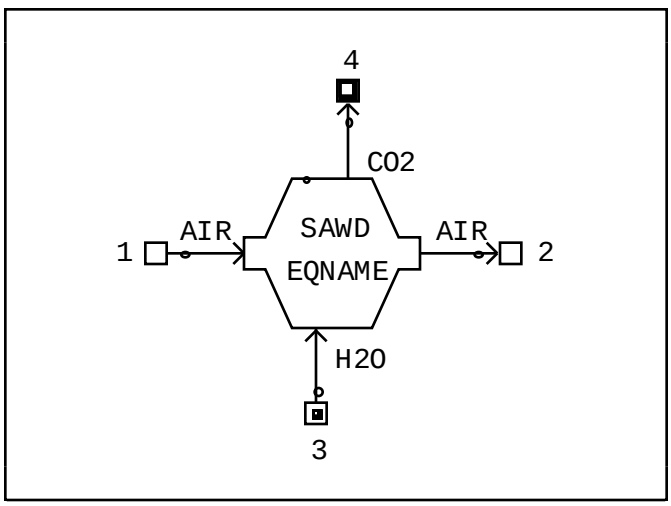
2. Welty, J.R., Wicks, C.E., and Wilson, R.E.: “Fundamentals of Momentum, Heat and Mass Transfer.” John Wiley & Sons, 1976.

3. Roshenow, W.M., Hartnett, J.P., and Ganic, E.N.: “Handbook of Heat Transfer Fundamentals.” McGraw-Hill Book Company, 1985.

4. McAdams, W.H.: “Heat Transmission.” McGraw-Hill Book Company, 1954.

10.39 SAWD—Solid Amine Water Desorbed Simulation Description

The SAWD routine simulates the transient carbon dioxide removal characteristics and steam desorb characteristics of the actual process. Time-averaged process flow rates and cabin heat loading will be provided as output if the system time step and the specified half-cycle durations are not compatible. Detailed two-bed simulation requires the system time step to be less than the cyclic intervals, and OPS logic is required to properly cycle air flow paths to the active component. Air is passed through the component via streams 1 and 2, where mass transfer rates are calculated using equilibrium isotherm data (i.e., bed loading and transient bed temperature). The routine is configured for steam desorb and thus inlet steam flow rate is a required parameter. Desorbed carbon dioxide leaves via stream 4. Condensed water vapor occurring during desorb evaporates during adsorb and leaves the component via stream 2.



10.39.1 Introduction

The SAWD has three main modes of operation; adsorption, desorption, and energy-transfer (E-T) assuming a two-bed subsystem. The E-T mode can be further subdivided into a passive energy-transfer phase and an active energy-transfer phase. Adsorption consists primarily of forced convection mass and heat transfer operations that, under normal conditions, include adsorption of CO_2 and desorption (bed drying) of H_2O. Desorption involves steam injection into the SAWD resulting in CO_2 expulsion from the bed. Passive E-T is very short in duration (approximately 20 s) and involves transfer of water vapor from bed-to-bed by joining the hot, high-pressure bed with the cold, low-pressure bed. Active E-T takes this water vapor transfer a step further by utilizing a fan downstream of the destination bed to "pull" the H_2O off the donor bed. A discussion of the analysis performed to verify the SAWD component can be found in reference 1.

10.39.2 General Assumptions

The following general assumptions were made in deriving the equations that are employed in the SAWD routine:

A. The temperature gradients within the adsorbent pellets are negligible.

B. For the analysis of mass and heat-transfer characteristics of a packed bed, it was assumed that gas and bed characteristics were uniform over any cross section of the bed normal to the gas flow direction, excluding intraparticle concentration gradients.

C Adsorption occurs by the diffusion of an adsorbate through the stagnant surface film at the exterior surface and then diffusion into the interior of the particle.

D. Heat of adsorption is independent of temperature and CO_2/H_2O concentration.

E. Thermal capacitance of the bed is independent of temperature and CO_2/H_2O concentration.

F. There is no interaction between the CO_2 and H_2O within the bed, i.e., their individual isotherms are independent of the other adsorbate concentration.

G. Hysteresis is ignored, i.e., it is assumed that adsorption and desorption follow the same isotherm path.

H. The effects of swelling of the amine bed are ignored.

10.39.3 Mass Balance

The mass transfer network developed is a modified version of that presented in reference 2. The approach is defined by the use of a bed

"element" of which a maximum of 20 can be arranged axially down the bed length. The actual number of elements is a user input constant. Each element contains three diffusion nodes and two arithmetic gas nodes as shown in figure 71. The SAWD component routine contains a mass transfer network for both CO_2 and H_2O. These networks allow for modeling of three separate mass transfer mechanisms: (1) gas-to-adsorbent particle surface, (2) intraparticle diffusion, and (3) interelement (axial) diffusion.

10.39.4 Mass Transfer to Adsorbent Surface

Since the adsorbate gas can be transferred only between the bed surface and the carrier gas, a vapor density network is utilized as shown in figure 73 for the calculation of gas-to-adsorbent surface CO_2 mass transfer. Using this method, CO_2 vapor densities are computed as a function of the inlet CO_2 concentration, packed-bed mass transfer convective effectivity, and the characteristic CO_2 equilibrium vapor density of the particle surface. All of the discussions here will address CO_2 mass transfer, but are also directly applicable to the H_2O mass transfer network.

The convective CO_2 mass transfer mechanisms from air-free stream to amine particle surface are simulated using the delta vapor density network. The convective effectivity associated with this network is computed with the following formula:

$$\varepsilon_m = 1 - e^{-(kA/\dot{v})} ,$$

where,

ε_m = mass transfer convective effectivity

k = overall packed bed convective mass transfer coefficient, ft/h

A = amine total particle surface area, ft^2

$\dot{v}$ = volumetric flow rate of air, ft^3/h.

The overall convective mass transfer coefficient is provided as a user-specified input constant. Its value is, however, a function of the local Reynolds, Sherwood, and Schmidt numbers and may be dynamically computed in the OPS logic if desired. Methodology behind the determination of packed bed transport properties may be obtained in references 3 and 4. Because of the complexity and uncertainty associated with this methodology, and because of available test data, this model will rely on parametric runs of the simulation to experimentally determine an average convective mass transfer coefficient. The volumetric air flow rate is calculated from inlet pressure, temperature, and total mass flowrate for the first axial element.

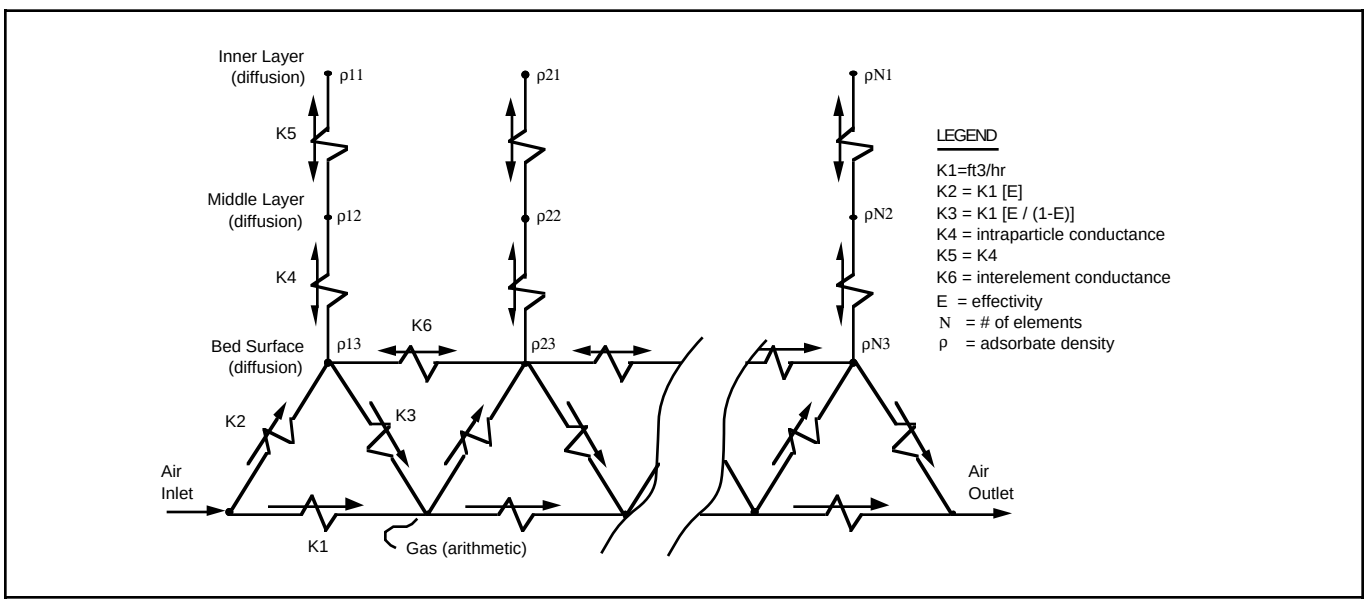


Figure 71. SAWD mass transfer network.

The calculation of the packed bed outlet CO_2 concentration in terms of the delta network relationships is given as follows:

$$\rho_{CO_2}^{out} = \frac{\rho_{CO_2}^{in} + [\varepsilon_m/(1+\varepsilon_m)]\rho_{eql}}{[1+\varepsilon_m/(1+\varepsilon_m)]},$$

where,

$\rho_{CO_2}^{in}$ = inlet CO_2 vapor density, lbm/ft^3

$\rho_{CO_2}^{out}$ = outlet CO_2 vapor density, lbm/ft^3

ρ_{eql} = bed equilibrium vapor density, lbm/ft^3.

These equations are calculated in series for each bed element. The CO_2 vapor density (ρ_{eql}) is calculated as discussed in the following paragraphs. The inlet CO_2 vapor density for subsequent elements is simply the outlet from the upstream delta network.

10.39.5 Intraparticle Mass Transfer

Because of the large thermal and concentration transients inherent in the adsorption/desorption process, it is theorized that an adsorbate gradient could exist within the individual adsorbent particles. This is caused by an intraparticle diffusion resistance that, at least partially, governs the overall rate of mass transfer to/from the particle. While there is no current evidence that this resistance is significant in the SAWD, the ability to model this phenomena will be available. The intraparticle mass transfer mechanism is analogous to heat transfer with a driving potential of equilibrium vapor density difference. The following equation is used to simulate this process.

$$\dot{m}_{CO_2} = K(\rho_i - \rho_j),$$

where,

K = diffusion mass transfer conductance, (ft^3/h)

ρi = equilibrium vapor density of node i, (lbm/ft^3).

The idealized adsorbent particle chosen was a sphere arbitrarily divided into three concentric layers. The adsorbent mass is assigned to the three layers as follows:

$$M_1 = \frac{M_2}{2} = \frac{M_3}{2},$$

where,

M_1 = mass of outer adsorbent layer, lbm

M_2 = mass of middle adsorbent layer, lbm

M_3 = mass of inner adsorbent layer, lbm.

Assuming the mass of the outer layer is located at the particle surface and the mass of the inner layers are located at a radius that subdivides the layer into two equal volumes, then the relationship between the layer overall conductances is:

$$K_{2-3} = \frac{1}{2.36} K_{1-2},$$

where,

$K_{2\text{-}3}$ = conductance between middle and inner layer (ft^3/h)

$K_{1\text{-}2}$ = conductance between outer and middle layer (ft^3/h).

In the mass transfer network, each particle layer is defined as a diffusion node (capacity for storing adsorbate). This capacity is dependent on the mass and temperature of the particle layer and the adsorbate gas pressure at the particle layer surface, and is generally defined in a three-dimensional function referred to as isotherm data. In order to provide a generalized analytical solution to the mass transfer equations, the isotherms can be linearized to provide for a localized constant capacitance term. This mass capacitance can be derived as follows.

The adsorbate mass loading as a percent of adsorbent bed weight can be expressed as:

$$\frac{M_a}{M_b} = f(PP, T),$$

where,

M_a = adsorbate mass, lbm

M_b = adsorbent bed mass, lbm

PP = adsorbate partial pressure (mmHg)

T = system temperature (°F).

Implicit differentiation results in the following linear partial differential equation:

$$\frac{dM_a}{dt} = M_b\left(\frac{df}{dPP}\frac{dPP}{dt} + \frac{df}{dT}\frac{dT}{dt}\right),$$

where,

t = time, h,

assigning capacitance terms as,

$$C_1 = M_b\frac{df}{dPP}\bigg|_T \simeq M_b\frac{\Delta f}{\Delta PP}\bigg|_T ,$$

$$C_2 = M_b\frac{df}{dT}\bigg|_{PP} \simeq M_b\frac{\Delta f}{\Delta T}\bigg|_{PP} .$$

For compatibility with the intraparticle mass transfer equations the capacitance, C_1, can be expressed in terms of equilibrium vapor density as follows;

$$C_1 = \frac{M_b}{R(T+460)}\frac{\Delta f}{\Delta \rho}\bigg|_T .$$

C_1 and C_2 represent the localized slope of the isotherm and isobar and can be calculated at small intervals around the current operating conditions (mass, partial pressure, and temperature), utilizing subroutines based on isotherm data for the amine adsorbent. It follows that for small time intervals:

$$\frac{dMa}{dt} \simeq C_1\frac{\Delta\rho}{\Delta t} + C_2\frac{\Delta T}{\Delta t} .$$

This allows a mass balance around each individual particle layer as follows:

$$\frac{dm_i}{d_t} = \sum_{j=1}^{n} K_{ij}\,(\rho_i - \rho_j) - \dot{m}_{\text{imp}} .$$

Combining equations results in:

$$(C_1/dt)(\rho_i - \rho_{i(o)}) + (C_2/dt)\,(T_i - T_{i(o)}) = \sum_{j=1}^{n} K_{ij}\,(\rho_i - \rho_j) - \dot{m}_{\text{imp}} ,$$

that can be utilized to calculate the equilibrium vapor density for each particle layer as follows:

$$\rho_i = \frac{(C_1/dt)\rho_{i(o)} + \sum_{j=1}^{n} K_{ij}\rho_j - (C_2/dt)\,(T_i - T_{i(o)}) - \dot{m}_{\text{imp}}}{(C_1/dt) + \sum_{j=1}^{n} K_{ij}} ,$$

where,

$\frac{dm_i}{d_t}$ = time rate of change of adsorbate mass on layer i, (lbm)

K_{ij} = conductance between layer i and adjacent layers j, (ft^3/h)

$\dot{m}_{\text{imp}}$ = mass flow imposed on layer from adjacent elements, (lbm/h)

ρ_i = equilibrium vapor density of layer i, (lbm/ft^3)

$\rho_{i(o)}$ = equilibrium vapor density of layer i at end of last time step, (lbm/ft^3)

T_i = temperature of layer i, (°F)

$T_{i(o)}$ = temperature of layer i at the end of the last time step, (°F)

n = total number of adjacent nodes.

The amount of mass adsorbed/desorbed for a given time step can then be calculated by:

$$\Delta m_a = C_1(\rho_i - \rho_{i(o)}) + C_2(T_i - T_{i(o)}) .$$

The SAWD amine CO_2 isotherm data were generated using equations from reference 5 and correlates well with the single isotherm given in reference 6. These data are based on the Langmuir equation that is used to describe monolayer,

chemical adsorption. The H_2O isotherm data were estimated using an arbitrary equation typical of multilayer, physical adsorption.

10.39.6 Interelement Mass Transfer

A third mass transfer mechanism, element-to-element axially within the bed, is also possible. During the adsorb half cycle, it is likely that the gas-to-particle mass transfer governs the axial movement of adsorbate down the bed and that the interelement mass transfer can be ignored. However, during the steam desorb and energy-transfer modes, the convective flow down the bed is relatively low and the interelement conductance may become a significant factor. The current configuration of the SAWD does not utilize this mass transfer mode. The mass transfer between elements is calculated by:

$$\dot{m}_{ij} = K_{ij}(\rho_i - \rho_j)\,,$$

where,

$\dot{m}_{ij}$ = mass transfer from element i to element j (lbm/h)

K_{ij} = mass transfer conductance from element i to element j, (ft^3/h)

ρ_i = equilibrium vapor density of surface layer of element i, (lbm/ft^3)

ρ_j = equilibrium vapor density of surface layer of element j, (lbm/ft^3).

10.39.7 Mass Balance (Desorption)

The element approach works extremely well for the adsorb mode when the carrier gas flow rate is near constant down the bed. During the steam desorb mode, the CO_2 flow rate down the bed varies from 0 to 1.2 lbm/h with no carrier gas present. A series flow down the bed similar to the adsorb network would be the most technically correct method for modeling the desorb mass transfer; however, the delta network used for the adsorb mode was developed for a configuration with inlet forcing functions of CO_2 partial pressure and carrier gas flow rate. As an approximation, an arbitrary (user-supplied) CO_2 desorption mass flowrate is implied uniformly to each element at the appropriate time through OPS logic. This flowrate may be timelined in OPS logic to provide a more realistic interface with the CO_2 accumulator. Steam consumption is arbitrary with the current SAWD model.

Since there is no evidence that the intra-particle mass transfer resistance is significant, the SAWD component arbitrarily places the entire bed mass on the outer-most particle layer to reduce computation time. If desired, modeling of the intraparticle mass transfer is calculated in the same manner as the adsorb mode.

Interelement mass transfer could be used to model the CO_2 “wave” formed during adsorption. Currently, the SAWD component does not utilize interelement mass transfer.

10.39.8 Mass Balance (Energy-Transfer)

As an approximation of the E-T mode, the initial H_2O bed loading for the adsorb mode (which immediately follows the E-T mode) is specified through OPS logic.

10.39.9 Energy Balance (Adsorb)

An R-C network approach is provided for the determination of the amine bed, canister, outer insulation, and bed free stream outlet temperatures. The thermal network is shown in figure 72. The convective delta technique is utilized to simulate the fluid-to-wall heat transfer mechanisms that typically dominate the overall heat transfer rates in a packed bed. Calculation of the thermal delta network effectivity is analogous to the mass transfer effectivity calculation and is expressed as:

$$\varepsilon_t = 1 - e^{-(hA/\dot{m}Cp)}\,,$$

where,

ε_t = thermal delta network effectivity

h = overall packed bed convective heat transfer coefficient, Btu/h ft^2 °F

$\dot{m}$ = mass flow rate of air, lbm/h

Cp = specific heat of air, Btu/lbm °F

A = total surface area of bed particles, ft^2.

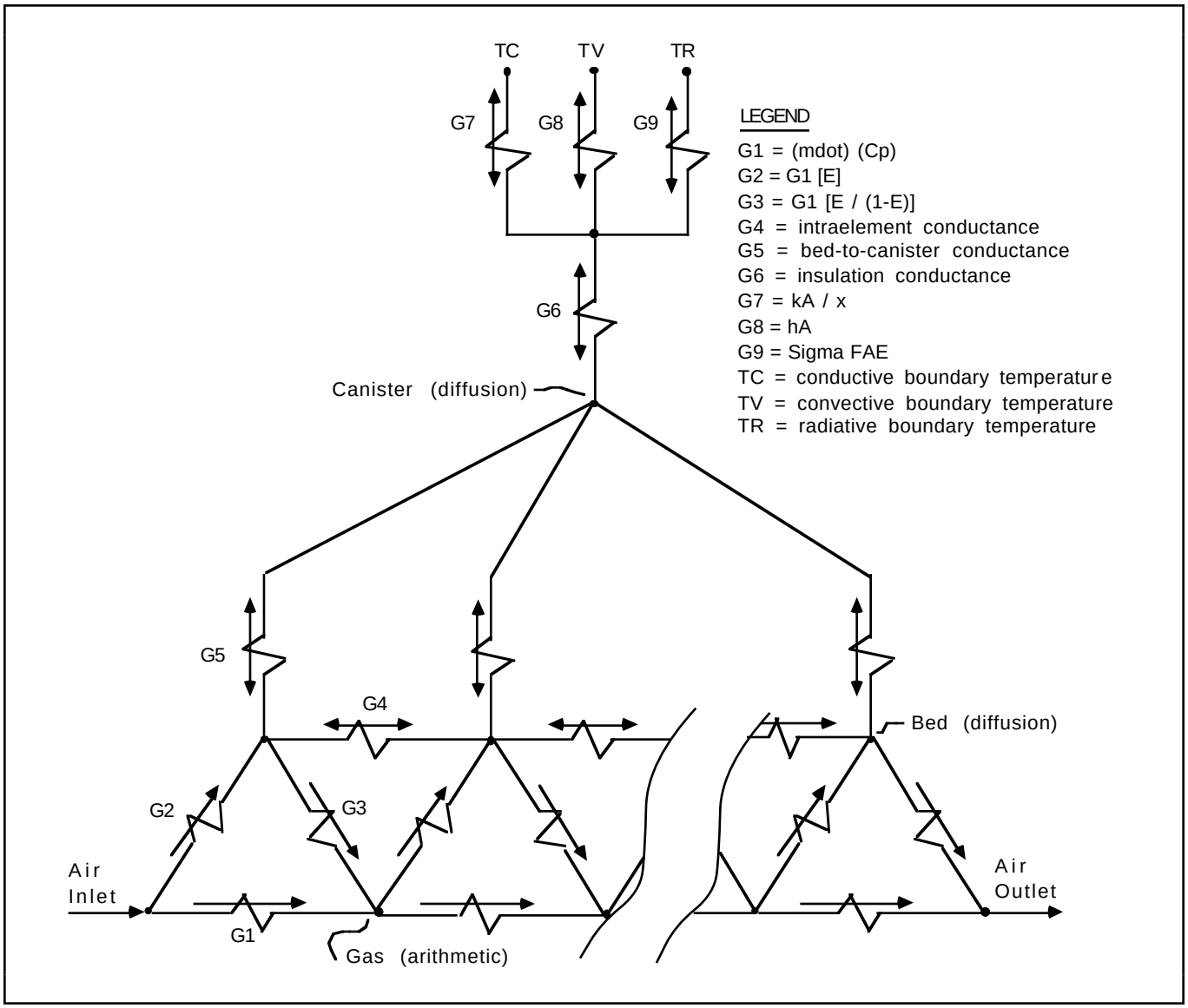


Figure 72. SAWD thermal network.

The overall heat transfer coefficient (h) is provided as a user-specified input constant. Its value may be dynamically computed (in the OPS logic) as a function of instantaneous Reynolds, Nusselt, and Prandtl numbers, if desired. The calculations for the bed element portion of the thermal network are included in a subroutine to the SAWD called TNETWK. The development of this routine is described in section 8 of this manual. Each bed element is tied to upstream and downstream elements as well as to the canister shell by linear conductors.

The following equation is utilized to calculate the temperature of each diffusion node:

$$T_i = \frac{(C/dt)\, T_{i(o)} + \sum_{j=1}^{n} G_{ij} T_j + \dot{Q}_{\text{imp}}}{C/dt + \sum_{j=1}^{n} G_{ij}},$$

where,

T_i = temperature of node (°F)

$T_{i(o)}$ = temperature of node at end of last time step, (°F)

G_{ij} = thermal conductance between node i and node j, (Btu/h/°F)

C = thermal capacitance, (Btu/°F)

dt = time step, (h)

$\dot{Q}_{imp}$ = imposed heat flow such as heat of adsorption, (Btu/h)

n = total number of adjacent nodes.

The arithmetic node temperatures are calculated by:

$$T_i = \frac{\sum_{j=1}^{n} G_{ij} T_j}{\sum_{j=1}^{n} G_{ij}} .$$

10.39.10 Energy Balance (Desorb)

Because of the simplifications required during the desorb mode, the only thermal consideration is the provision for a user-input, desorbed CO_2 outlet temperature. This temperature may be timelined in OPS logic to achieve a realistic interface with the CO_2 accumulator.

10.39.11 Energy Balance (Energy-Transfer)

The only thermal consideration for the E-T mode is the provision for setting the initial bed temperature for the adsorb mode (which immediately follows the E-T mode) through OPS logic.

10.39.12 Pressure Drop

The pressure drop through the SAWD during the adsorb cycle is calculated based on user specified equivalent lengths and diameters in the subroutine PIPEDP discussed in section 8.

10.39.13 Input Parameters

The input parameters for the SAWD are presented in table 66.

Table 66. SAWD edit screen.

```
CASE NAME: DEFAULT                                  SAWD EDIT SCREEN                                  LAST UPDATED: 890526
SUBSYSTEM:                COMPONENT:                                                                  RECORD NUMBER: 000001
_________________________________________________________________________________________________________________________
|       ** GENERAL INPUT DATA **          |                                                |       ** PRESSURE DROP PARAMETERS **        |
| 19 TOT PARTCL SURF AREA (FT2)   20.00   | 25 OP MODE FLAG  (0-A,1-D,2-ETA,3-ETD) 1.0     | 35 AIR STREAM EQUIV LENGTH     (FT)    1.00 |
| 29 CANISTER WEIGHT       (LB)   10.00   | 12 INIT BED/CANISTER TEMP      (F)  100.00     | 36 AIR STREAM EQUIV DIAMETER   (IN)   6.000 |
| 28 BED WEIGHT            (LB)    8.50   | 10 INIT CO2 BED LOADING       (LB)  0.2500     |       ** THERMAL INTERFACE DATA **          |
| 13 # OF AXIAL ELEMENTS (<=20)    10.0   | 11 INIT H2O BED LOADING       (LB)  1.5000     | 22 EXTERNAL CONVECTIVE G (BTU/HR/F)    4.00 |
|                                         | 26 HEAT OF ADSORPTION (BTU/LB-CO2)  500.00     | 23 MOUNTING STRUCTURE G  (BTU/HR/F)   10.00 |
| 17 CO2 INTRAPARTCL G (FT3/HR)     0.0   | 27 HEAT OF ADSORPTION (BTU/LB-H2O)  500.00     | 24 EXTERNAL F-A-E             (FT2)    5.00 |
| 18 H2O INTRAPARTCL G (FT3/HR)     0.0   | 15 CO2 DIFFUSIVITY/FT      (FT/HR)   500.0     | 20 INSULATION G          (BTU/HR/F)    2.00 |
|                                         | 16 H20 DIFFUSIVITY/FT      (FT/HR)   500.0     | 21 BED-CANISTER G        (BTU/HR/F)   25.00 |
| 37 INTERELEM THERM G (BTU/HR/F)   0.0   | 14 FREE STREAM H    (BTU/HR/FT2/F)   75.00     |   ** SUB-SOLUTION CONTROL PARAMETERS **     |
|                                         |          ** RESOURCE TRACKING **               | 33 MINISTEP MAX ITERATIONS              25.0 |
| 38 CO2 DESP OUTLET TEMP   (F)   75.00   | 9 VOLUME                     (FT3)    0.00     | 32 MINISTEP DIVISIONS                   10.0 |
| 39 CO2 DESP FLOW RATE (LB/HR)    1.00   | 8 WEIGHT                      (LB)    0.00     | 31 MINISTEP DAMPING FACTOR   (0<D1) .50     |
| 40 E-T MODE H2O MDOT  (LB/HR)    0.00   | 7 POWER                        (W)    0.00     | 30 MINISTEP RELAXATION            0.100E-04 |
| 41 BED HEAT CAPACT (BTU/LB/F)    0.50   |                                                | 34 DISPLAY MINISTEP ITERATONS (N=0,Y=1) 0.0 |
|_________________________________________|________________________________________|________________________________________________|
|                              ** OUTPUT DATA **                                  | ** BENCHMARK DATA **  MIN         NOM         MAX    |
| 43 CURRENT CO2 BED LOADING (%)    0.00 45 AVG BED TEMP        (F)    0.00      | AIR IN     (LB/HR)  0.00E+00    0.00E+00    0.00E+00 |
| 44 CURRENT H2O BED LOADING (%)    0.00                                         | H2O IN     (LB/HR)  0.00E+00    0.00E+00    0.00E+00 |
| 48 TOTAL C02 REMOVED      (LB)    0.00                                         |                                                    |
|_________________________________________________________________________________|________________________________________________|
```

References

1. "Verification Analysis for the Computer Aided System Engineering and Analysis (CASE/A) Solid Amine Water Desorbed (SAWD) Component Routine," McDonnell Douglas Space Systems Company Report No. MDC W5180, May 1989.

2. Blakely, R.L., and Taylor, B.N.: "Computer Analysis Techniques for Spacecraft Adsorption Beds." AIChE, National Meeting, 64th Symposium on Adsorption-Part III, 1969, Paper 19C (A69-21931).

3. Kunii, D., and Suzuki, M.: "Particle-to-Fluid Heat and Mass Transfer in Packed Beds of Fine Particles." Int. J. Heat Mass Transfer, vol. 10, 1967.

4. Ruthven, D.M.: “Principles of Adsorption and Adsorption Processes.” John Wiley and Sons, 1984.

5. Holland, C.D., and Liapis, A.I.: “Computer Methods for Solving Dynamic Separation Problems.” McGraw-Hill Book Co., 1983.

6. “SAWD Operation Manual,” United Technologies/Hamilton Standard Report No. TDH-DI-06, MMC-ECLSS-7, April 1987.

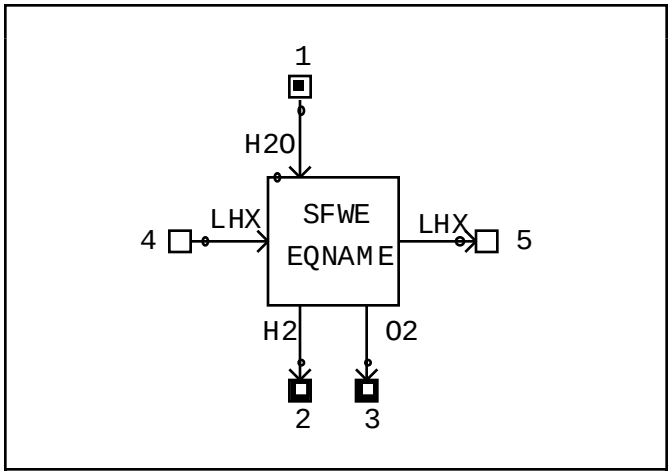
10.40 SFWE—Static Feed Water Electrolysis Simulation Description

The SFWE routine is designed to simulate the operation of the SFWE method for the production of O_2 and H_2 from water. The reactions that take place in the SFWE are governed by the following equations:

$$\text{Anode: } 4OH^- \text{--->} \; O_2 + 2H_2O + 4e^-$$

$$\text{Cathode: } 4H_2O + 4e^- \text{--->} \; 2H_2 + 4OH^-$$

$$\text{Overall: } 2H_2O + \text{Power--->} \; 2H_2 + O_2$$



10.40.1 General Discussion

The SFWE routine simulates O_2 production from an aqueous electrolyte solution of potassium hydroxide (KOH). Makeup water enters the process through an internal fluids control assembly that regulates the operating conditions within the electrolysis cell. Initially, both the water feed cavity and cell electrolyte matrix (internal cell sections) are assumed to have equal concentrations of electrolyte. As electrical power is applied to the electrodes, water is electrolyzed from the cell matrix, creating a concentration gradient between the electrolyte in the feed cavity and electrolyte in the cell matrix. Water vapor diffuses from the water feed cavity into the cell matrix due to this concentration difference. Waste heat within the cell is removed by the continuous flow of the feed cavity stream through the cell matrix. The recirculating fluid then rejects heat to an external cooling loop via a liquid/liquid HX. A detailed description of the routine mass and thermal balances are outlined in the following paragraphs.

10.40.2 Mass and Voltage Control Relationship

H_2, O_2, and H_2O electrolysis rates are calculated based on the governing chemical equations and the Faradaic efficiency (FE) of the cell. The desired oxygen production rate is set by the user, and the current input required to obtain this rate is determined by the following equation:

$$\text{current} = \frac{\dot{m}_{O_2}\left(1519.7 \dfrac{\text{Amp}\bullet\text{h}}{\text{lb O}_2}\right)}{\text{FE}} .$$

The H_2 production and H_2O electrolysis rates are then determined based on this current value using the following equations:

$$\dot{m}_{H_2} = \text{FE}\left(\frac{\text{cell current}}{12062.0 \dfrac{\text{Amp}\bullet\text{h}}{\text{lb H}_2}}\right),$$

$$\dot{m}_{H_2O} = \text{FE}\left(\frac{\text{cell current}}{1349.9 \dfrac{\text{Amp}\bullet\text{h}}{\text{lb H}_2\text{O}}}\right).$$

These equations were derived from the balanced cell reaction equations. At startup the SFWE subsystem monitors cell voltage, and when cell voltage falls below a specified maximum voltage, the current input to the cell is incremented until the required current is reached. The initial current input to the cell and a current increment are supplied by the user. During steady-state operation, the current input to the cell is maintained at the rate required to obtain the desired O_2 production rate. Cell voltage is calculated at each controller time step using the following equation supplied by reference 1.

$$V = 1.25 + 0.2287\left(\frac{CD}{100}\right) - 0.07625\left(\frac{CD}{100}\right)^2 + 0.001583\left(\frac{CD}{100}\right)^3 - 0.00125\left(\frac{CD}{100}\right)^4 - 0.002(T-180) ,$$

where,

CD = current density $\left(\frac{\text{Amps}}{\text{ft}^2}\right)$

T = cell temperature (°F)

This equation correlates cell voltage to temperature over a wide range of current density. However, the valid temperature range of the equation is limited to cell temperatures that occur only during steady state operation (137 °F to 140 °F). Due to the limited temperature range of the above equation, the coefficients of the voltage function used in the simulation are provided as input parameters. Input positions for the coefficients are provided in the curve fit coefficients subscreen to the SFWE main edit screen as shown in table 67.

10.40.3 Thermal Calculations

The thermal network used to model the SFWE subsystem consists of the TNETWK subroutine, available in CASE/A, attached to an internal coolant bypass loop and an external liquid/liquid HX. Figure 73 shows a schematic of the thermal network used in the SFWE subroutine. TNETWK consists of the coolant delta network attached to the coolant wall, mass, shell, and environment nodes. The cell mass node represents everything enclosed by the polysulfone frame within the cell stack. This area includes all electrolysis cells (anodes, cathodes, and electrolyte) along with the gas and feed cavities. Capacitances and conductances are input based on the characteristics of this module. The delta network conductances input to the TNETWK subroutine are based on the coolant control assembly (CCA) internal coolant mass flow rate to the electrolysis module. More details on the TNETWK routine can be found in the TNETWK description in section 8.3.3.

The method used for determining the heat produced by the reaction required the use of the first law of thermodynamics for a reacting system. From reference 2 the first law for a reacting system is given by:

$$\dot{Q}+\sum_{R} n_i\left(\bar{h}_f^o+\Delta\bar{h}\right)_i=\dot{W}+\sum_{P} n_e\left(\bar{h}_f^o+\Delta\bar{h}\right)_e$$

where,

n = molar flow rate $\left(\frac{\text{lb-mol}}{\text{hr}}\right)$

$\bar{h}_f^o$ = enthalpy of formation at standard state $\left(\frac{\text{Btu}}{\text{lb-mol}}\right)$

$\Delta\bar{h}$ = change in enthalpy of formation with temperature $\left(\frac{\text{Btu}}{\text{lb-mol}}\right)$

$\dot{W}$ = work input to the system $\left(\frac{\text{Btu}}{\text{h}}\right)$

$\dot{Q}$ = resultant heat load on the module $\left(\frac{\text{Btu}}{\text{h}}\right)$

The subscripts R and P refer to the reactants and products, respectively. The multiple of the molar flow rate and enthalpy of formation is carried out over the summation of the reactants and products. The i and e subscripts refer to the inlets and exits of the system under analysis. Applying the equation above to an electrolysis cell, the separate terms in the first law equation are evaluated as follows:

Table 67. SFWE curve fit edit screen.

CASE NAME: DEFAULT — SFWE CURVE FIT COEFFICIENTS EDIT SCREEN — LAST UPDATED: 890215
SUBSYSTEM: COMPONENT: — RECORD NUMBER: 000001

DEPENDENT VARIABLE	INDEPENDENT VARIABLE	^3	^2	^1	^0
O2 DELTA ENTHLPY (BTU/LB-MOL)	TEMPERATURE (deg R)	0.344435E-06	-.224671E-03	6.98336	-3735.61
H2 DELTA ENTHLPY (BTU/LB-MOL)	TEMPERATURE (deg R)	0.188078E-05	-.246319E-02	7.54886	-3625.18
H2O VAP DELTA ENTHLPY (BTU/LB-MOL)	TEMPERATURE (deg R)	0.179945E-06	0.176168E-04	7.88054	-4261.91
DEPENDENT VARIABLE: V	**INDEPENDENT VARIABLE**	**^4**	**^3**	**^2**	**^1**
CELL VOLTGE EQUATION COEFFCNTS:	i/100 (ASF)	-.125000E-02	0.158300E-01	-.762500E-01	0.228700
V = V(i,CELL TEMP)	CELL TEMP (degF)	0.000000E+00	0.000000E+00	0.000000E+00	-.200000E-02
(i = CELL CURRENT DENSITY)	^0	1.61000			

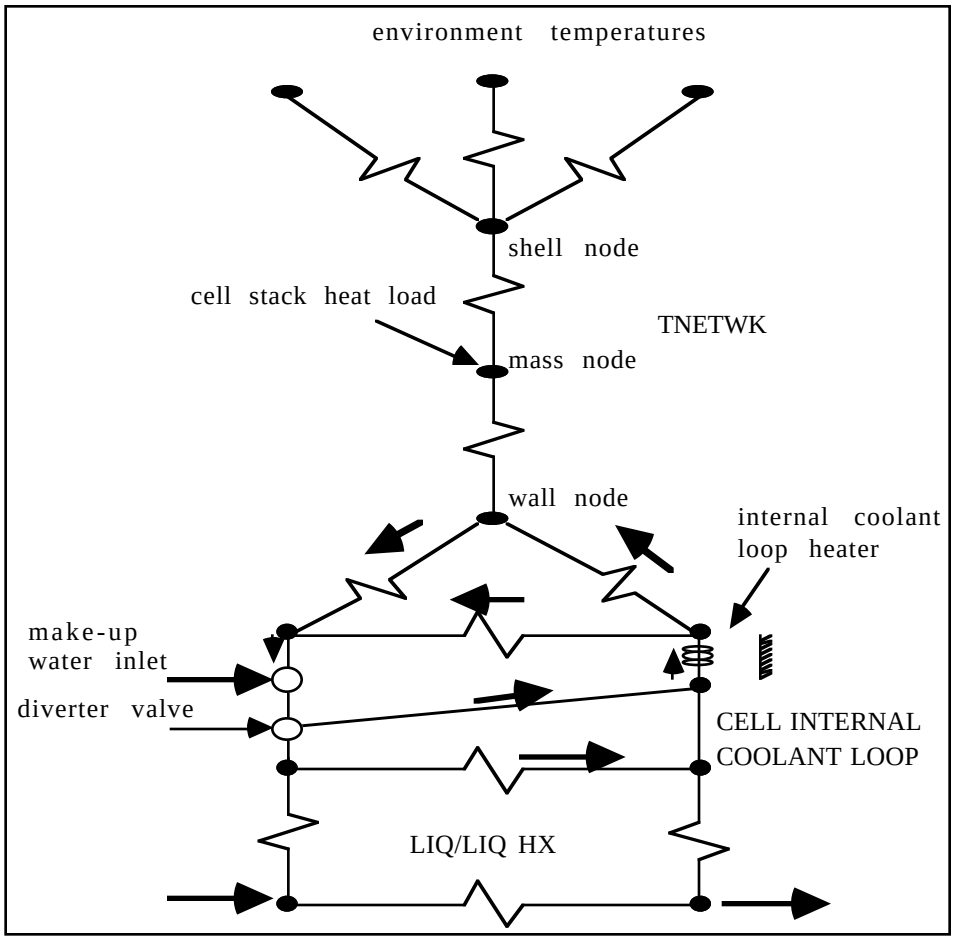


Figure 73. SFWE thermal network schematic.

$$\sum_R n_i\left(\bar{h}_f^o + \Delta\bar{h}\right)_i = n_{H_2O}\left(-104036.0+\Delta\bar{h}\right),$$

$$\sum_P n_e\left(\bar{h}_f^o+\Delta\bar{h}\right)_e = n_{H_2}\Delta\bar{h}+n_{O_2}\Delta\bar{h},$$

$$\dot{W} = V*I*3.41214\left(\frac{\text{Btu}}{\text{h-Watt}}\right).$$

V and I are the voltage and current of the cell, respectively. The heat load on the cell due to the reaction is then given by:

$$\dot{Q} = \text{VI}3.41214 - n_{H_2}\Delta\bar{h} - n_{O_2}\Delta\bar{h} + n_{H_2O}\left(-104036.0+\Delta\bar{h}\right).$$

This is the heat produced by the exothermic reaction at the cell core. The signs were changed in the above equation because the first law analysis of the reactants and products will show a negative heat load for an exothermic reaction. This heat load is the heat input to the cell stack diffusion node; therefore, the sign must be changed. The heat produced, as determined by the above equation, is multiplied by the number of cells in the electrolysis module and then passed to the TNETWK subroutine as the mass node heat load.

The CCA is used to control cell stack temperature and supply makeup feedwater to the individual cells. Cell stack temperature is controlled by regulating the cell stack inlet temperature of the CCA fluid to a specified temperature band. The internal coolant loop was developed to model the temperature control characteristics of the CCA and the replacement water feed from the fluids control assembly (FCA). The internal coolant loop consists of the replacement water stream from the FCA, diverter valve, and heater. The diverter valve is used to control the amount of CCA fluid that exchanges heat with the external HX. Internal coolant bypass is controlled to a range specified by the user. The increment at which the diverter valve bypass is adjusted is also a user input. Bypass valve

position is adjusted if the coolant temperature at the module inlet falls outside the user-specified operating temperature bandwidth. A maximum bypass of 1.0 corresponds to a total bypass of internal coolant flow from the external liquid/liquid HX, such that no heat exchange takes place with the external loop. The internal loop electrolysis module inlet temperature is calculated based on a first-law analysis of the bypass stream and the fluid that is not bypassed by the diverter valve. The bypass fraction is defined as:

$$BP = \left(\frac{\dot{m}_h}{\dot{m}_c}\right),$$

where $\dot{m}_h$ is the flow rate of the fluid bypassing the external cooling loop and $\dot{m}_c$ is the internal loop total mass flow rate. The bypassed stream flow rate is then given as:

$$\dot{m}_h = BP\dot{m}_c \,.$$

Flow that does not bypass the external loop is determined by:

$$\dot{m}_e = \dot{m}_c - \dot{m}_h = \dot{m}_c - BP\dot{m}_c = \dot{m}_c(1.0 - BP)$$

A first-law analysis of the streams as they mix after exchanging heat with the external HX is carried out as follows:

$$\dot{Q} + \sum \dot{m}_i\left(\mathrm{h}_i + \frac{v_i^2}{2g_c}\right) = \dot{W} + \sum \dot{m}_e\left(h_e + \frac{\mathrm{v}_e^2}{2g_c}\right).$$

Assuming that the differences in velocity are small, and with no heat transfer or work from the streams, the following equation is obtained:

$$\sum \dot{m}_i h_i = \sum \dot{m}_e h_e \,,$$

$$\sum \dot{m}_i h_i = BP\dot{m}_c c_p T_w + \dot{m}_c(1.0 - BP)c_p T_x \,.$$

T_x is the temperature leaving the liquid/liquid HX and T_w is the temperature of the coolant leaving the electrolysis module, after it has mixed with the replacement water stream.

$$\sum \dot{m}_e h_e = \dot{m}_c c_p T_{in} \,.$$

Changes in c_p over the variations in the temperature between T_x and T_w will be small, therefore T_{in} is determined by:

$$\dot{m}_c T_{\mathrm{in}} = \mathrm{BP}\dot{m}_c T_w + \dot{m}_c(1.0 - BP)T_x \,,$$

$$T_{\mathrm{in}} = BPT_w + (1.0 - BP)T_x \,.$$

Mixing of the replacement water stream with the internal loop coolant was analyzed following the same procedure, except that the differences in specific heat between the mixing streams were taken into consideration. The following equation was obtained:

$$T_w = \frac{\left(\dot{m}_c - \dot{m}_r\right)c_p T_{\mathrm{out}} - \dot{m}_r T_r}{\dot{m}_c c_p} \,.$$

The r subscript signifies replacement stream properties. $\mathrm{T}_{\mathrm{out}}$ is the temperature of the coolant as it leaves the electrolysis module and c_p is the specific heat of the coolant.

The heater in the internal coolant loop was used to heat the internal loop fluid before it re-entered the electrolysis module. TNETWK was used to model the internal loop heater, with the heat input for TNETWK being the heater power. T_{in}, as defined above, was input as the initial mass node temperature and delta network inlet temperature. The actual electrolysis module inlet temperature will be the value for T_{out} determined by the TNETWK (heater model) subroutine. Fluid capacitance input for the TNETWK (heater model) mass node was calculated as follows:

$$cap_c = \dot{m}_c c_{p_c} \Delta t \,,$$

where Δt is the time increment.

The external cooling loop is a parallel flow liquid/liquid HX that is modeled by using the effectivity/NTU method of reference 3. The effectivity of the HX is determined by:

$$\varepsilon = \frac{1.0 - e^{-NTU(1.0 + C_{min}/C_{max})}}{\left(1.0 + \frac{C_{min}}{C_{max}}\right)} .$$

NTU is a dimensionless parameter and is defined by:

$$NTU = \frac{UA}{C_{min}} ,$$

where UA is the HX conductivity multiplied by the effective area. The HX value for UA must be input by the user. C_{min} is the capacity rate of the hot fluid ($\dot{m}_e c_{p_c}$). In this case, the fluid represented here is the fluid not bypassed by the diverter valve. The capacity rate of the hot fluid is then given by:

$$C_{min} = \dot{m}_c (1.0 - BP) c_{p_c} ,$$

where BP is the position of the bypass diverter valve and c_{p_c} is the specific heat of the coolant. C_{max} is the heat capacity rate ($\dot{m} c_p$) of the cold fluid. Following the procedure given in reference 3, the hot fluid outlet temperature is given by:

$$T_x = \frac{T_w \left(Y + \frac{C_{min}}{C_{max}}\right) + (1.0 - Y) T_z}{\left(1.0 + \frac{C_{min}}{C_{max}}\right)} ,$$

where Y is defined as,

$$Y = 1.0 - \varepsilon - \left(\frac{C_{min}}{C_{max}}\right)\varepsilon .$$

The cold fluid outlet temperature is then given by:

$$T_y = T_x + Y(T_z - T_w) .$$

T_z and T_y are the cold fluid inlet and outlet temperatures, respectively.

10.40.4 Pressure Calculations

The inlet stream pressure must be specified by the upstream component, which supplies the module with water. Outlet O_2 and H_2 stream pressures are specified in the routine by passing a value of 0.0 lb/in^2 absolute pressure for streams 2 and 3 to the PSPEC routine. This allows the pressures in the outlet streams, downstream of the O_2 and H_2 cavity relief valves, to be set by the downstream component. The relief valves maintain cavity pressures at the user-specified level. Pressure drop for the external coolant loop stream is accomplished through the use of the PIPEDP routine. The user must input the equivalent length and equivalent diameter of the external liquid coolant loop passage.

10.40.5 Input Parameters

The user input parameters for the SFWE are shown in tables 67 and 68 and the outputs and benchmarks are shown in table 69.

References

1. Life Systems Inc., Static Feed Water Electrolysis Technical Information.

2. Van Wylen and Sonntag: “Fundamentals of Classical Thermodynamics.” Second edition, John Wiley & Sons Inc., 1973, pp. 690–697.

3. White: “Heat and Mass Transfer.” Addison-Wesley Publishing Company, 1988, pp. 597–598.

Table 68. SFWE main edit screen.

```
CASE NAME: DEFAULT                                  SFWE EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                 RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|        ** GENERAL INPUT DATA **          |                                                      |                                |
| 10 OXYGN PRDCTN RT (LB O2/HR)  0.2300    |    ** INTERNAL COOLING LOOP **                       |    ** TNETWRK DATA **     7 POWER  (WATTS)      0.00 |
| 16 FARADAIC EFFICIENCY  (0-1)    1.00    | 13 EFFECTIVITY            0.99000                    | 11 MAX # LOOPS   100.0    8 WEIGHT   (LBM)      0.00 |
| 17 NUMBER OF ACTIVE CELLS       12.00    | 15 FLOW RATE     (LB/HR)  238.00                      | 12 RELAXATION 0.000010    9 VOLUME  (FT^3)      0.00 |
| 18 WEIGHT PERCENT KOH           27.00    | 19 SET POINT TEMP     (F)  133.00                     |                                                        |
| 25 EFFECTIVE CELL AREA (FT^2)  0.2500    | 33 INITIAL TEMP       (F)   75.00                     |           ** CELL START-UP PROPERTIES **               |
| 28 ELECTROLYTE WIDTH     (FT) 0.00125    | 36 TEMP BANDWIDTH     (F)    4.00                     | 35 N2 PURGE MODE OFF TIME                (HR)      0.90 |
| 29 O2 CAVITY VOLUME    (IN^3)  0.7470    | 37 INITIAL BYPASS             1.00                     | 42 H2 CAVITY PURGE PRESSURE          (PSIA)    165.00 |
| 30 H2 CAVITY VOLUME    (IN^3)  0.7270    | 38 MAX BYPASS                 1.00                     | 43 O2 CAVITY PURGE PRESSURE          (PSIA)    167.00 |
| 32 INITIAL CELL TEMP      (F)   75.00    | 39 MIN BYPASS                 0.50                     | 44 INITIAL CELL CURRENT              (AMPS)      8.00 |
| 34 CNTRLLR SMPLE TME STP (HR) 0.00281    | 41 HEATER PWR   (BTU/HR)  170.00                       | 45 CURRENT INCREMENT                 (AMPS)      0.50 |
| 40 MAX VOLTAGE PER CELL          1.70    | 46 SPCFC HT   (BTU/HR-F)  0.8000                       |                                                        |
| 69 SHUT DOWN TIME        (HR)    0.00    | 68 BYPASS INCREMENT      0.08000                       |           ** THERMAL NETWORK PROPERTIES **             |
| 70 OPERATION FLAG                0.00    |                                                        | 20 FORM FACTOR*AREA*EMISSIVITY      (FT^2)    0.5000 |
|                                          |          ** EXTERNAL LLHX **                           | 21 SHELL-ENVIRONMENT CONDUCTION (BTU/HR-F)      2.50 |
| ** STEADY-STATE OPERATING PRESSURES **   | 14 UA         (BTU/HR-F)    0.90                     | 22 SHELL-ENVIRONMENT CONVECTION (BTU/HR-F)   0.50000 |
| 26 O2 CAVITY           (PSIA)  182.00    | 47 EQUIV LENGTH     (FT)    5.00                     | 23 MASS-SHELL NODE  CONDUCTANCE (BTU/HR-F)      5.00 |
| 27 H2 CAVITY           (PSIA)  180.00    | 48 EQUIV DIAMETER   (IN)   0.500                     | 24 WALL-MASS NODE CONDUCTANCE   (BTU/HR-F)   5000.00 |
|                                          |                                                        | 31 CELL THERMAL CAPACITANCE        (BTU/F)    1.2000 |
| ** CURVE FIT COEFFCNTS (^L TO OPEN) _    | ** OUTPUT/BNCHMRK (^L TO OPEN) _                       |                                                        |
|__________________________________________|________________________________________________________|________________________________________________________|
```

Table 69. SFWE outputs edit screen.

```
CASE NAME: DEFAULT                                  SFWE OUTPUTS EDIT SCREEN                          LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                 RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                    ** OUTPUT DATA **                     | ** BENCHMARK DATA **          MIN           NOM           MAX |
| *************** O2 OUTLET PROPERTIES *****************    | H2O RATE IN   (LBM/HR)       .00000E+00    .00000E+00    .00000E+00 |
| 72 FLOW RATE                    (LBM/HR)     0.000000    | H2 STREAM OUT (LBM/HR)       .00000E+00    .00000E+00    .00000E+00 |
| 73 O2 MASS FRACTION                           0.00000    | O2 STREAM OUT (LBM/HR)       .00000E+00    .00000E+00    .00000E+00 |
| 74 N2 MASS FRACTION                           0.00000    |____________________________________________________________________|
| 75 H2O MASS FRACTION                          0.00000    | 80 INTERNAL COOLANT INLET TEMPERATURE (F)       0.00000            |
| *************** H2 OUTLET PROPERTIES *****************    | 81 CELL CURRENT                     (AMPS)      0.00000            |
| 76 FLOW RATE                    (LBM/HR)     0.000000    | 82 INDIVIDUAL CELL VOLTAGE                      0.00000            |
| 77 H2 MASS FRACTION                           0.00000    | 83 COOLANT BYPASS FRACTION                      0.00000            |
| 78 N2 MASS FRACTION                           0.00000    | 84 LLHX OUTLET TEMPERATURE            (F)       0.00000            |
| 79 H2O MASS FRACTION                          0.00000    | 85 O2 STREAM OUT                (LBM/DAY)       0.00000            |
|                                                          | 86 H2 STREAM OUT                (LBM/DAY)       0.00000            |
|                                                          | 87 ACTUAL CELL TEMP                   (F)       0.00000            |
|__________________________________________________________|____________________________________________________________________|
```

10.41 SINK—SINK Simulation Description

This routine simulates the performance of an ideal sink (SINK). The principal user input is the SINK inlet pressure. If the pressure is zero, then the SINK will accommodate the upstream pressure. If the pressure is nonzero, then the SINK will simulate a back-pressure of the specified magnitude. It should be noted that only one stream can be connected to the SINK inlet. Use of a SUM component upstream of the SINK is required if more than one stream is to flow into the SINK. The SINK will track the total accumulated mass. The SINK data base parameters are shown in table 70.

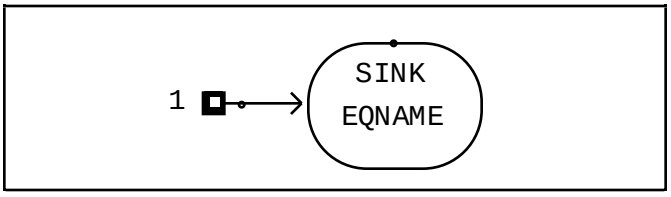
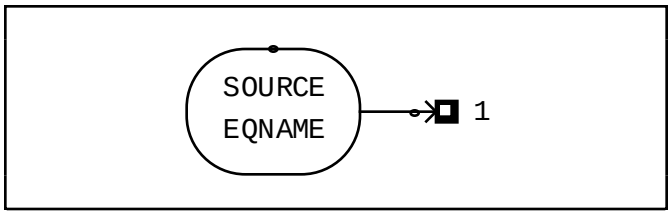


Table 70. SINK edit screen.

```
CASE NAME: DEFAULT            SINK EDIT SCREEN              LAST UPDATED: 890215
SUBSYSTEM:                    COMPONENT:                   RECORD NUMBER: 000001
_____________________________________________________________________________
|       ** GENERAL INPUT DATA **      |                                      |
| 10 BACK PRESSURE     (PSIA)    0.00 |  ** RESOURCE TRACKING **             |
|                                     | 7 POWER  (WATTS)    0.00             |
|                                     | 8 WEIGHT   (LBM)    0.00             |
|                                     | 9 VOLUME  (FT^3)    0.00             |
|                                     |                                      |
|_____________________________________|______________________________________|
|                ** BENCHMARK DATA **                |    ** OUTPUT DATA **    |
|                       MIN        NOM        MAX   |13 ACCM MASS,Lb 152.477448|
| INLET FLOW (LB/HR)  .00110513  .39706624  0.73    |14 FLOW IN,Lb/H 0.278     |
| INLET TEMP (DEG,F)  41.219559  43.345890 48.099625|15 TEMP IN,DegF    41.5   |
| INLET PRESS (PSIA)   0.10E-02   0.10E-02  0.10E-02|16 PRES IN,Psia  0.100E-02|
|_____________________________________________________|_________________________|
```

10.42 SOURCE—SOURCE Simulation Description

This routine simulates the performance of an ideal fluid source (SOURCE). The SOURCE provides the downstream component with a user-specified pressure, temperature, mass flowrate, and fluid composition.



If the source pressure is set to zero, then it will match the downstream configuration pressure characteristics. The SOURCE component may be substituted for a STORE/PUMP combination. The SOURCE primary data base parameters are shown in table 71. The SOURCE constituent data base parameters are shown in table 72. In the latter table the constituents from the ED LABELS screen in figure 23 have been substituted for CONSTIT 2 through CONSTIT 9 with the MASS FRACTIONs of each specified.

Table 71. SOURCE main edit screen.

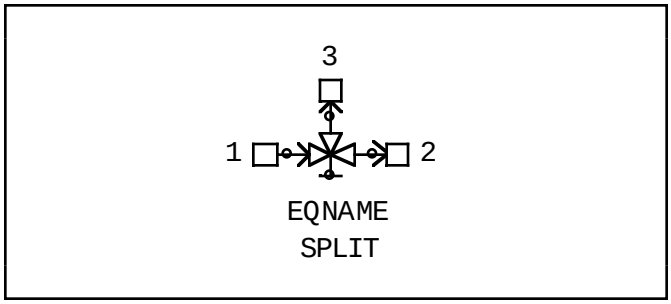
```
CASE NAME: DEFAULT            SOURCE EDIT SCREEN          LAST UPDATED: 890215
SUBSYSTEM:                   COMPONENT:                 RECORD NUMBER: 000001
_____________________________________________________________________________
|            ** GENERAL INPUT DATA **                |  ** RESOURCE TRACKING **  |
| 10 SOURCE PRESSURE          (PSIA)    14.7000 | 7 POWER   (WATTS)    0.00   |
| 11 SOURCE TEMPERATURE             (F)   75.00 | 8 WEIGHT    (LBM)    0.00   |
| 12 SOURCE FLOWRATE          (LBM/HR)    0.00 | 9 VOLUME   (FT^3)    0.00   |
|                                                |                             |
| FLUID CONSTITUENT DATA            (^L TO OPEN)_|                             |
|_______________________________________________|_____________________________|
```

Table 72. SOURCE constituent edit screen.

```
CASE NAME: DEFAULT       SOURCE CONSTITUENT SCREEN        LAST UPDATED: 890215
SUBSYSTEM:               COMPONENT:                      RECORD NUMBER: 000001
_____________________________________________________________________________
|    LABEL    |    MASS     |    LABEL    |    MASS     |    LABEL    |    MASS     |
|             |  FRACTION   |             |  FRACTION   |             |  FRACTION   |
|_____________|_____________|_____________|_____________|_____________|_____________|
| OXYGEN      1.0000000 | CONSTIT 19   0.0000000 | CONSTIT 36   0.0000000 |
| NITROGEN    0.0000000 | CONSTIT 20   0.0000000 | CONSTIT 37   0.0000000 |
| CARBON DIOX 0.0000000 | CONSTIT 21   0.0000000 | CONSTIT 38   0.0000000 |
| HYDROGEN    0.0000000 | CONSTIT 22   0.0000000 | CONSTIT 39   0.0000000 |
| CARBON (S)  0.0000000 | CONSTIT 23   0.0000000 | CONSTIT 40   0.0000000 |
| METHANE     0.0000000 | CONSTIT 24   0.0000000 | CONSTIT 41   0.0000000 |
| WATER       0.0000000 | CONSTIT 25   0.0000000 | CONSTIT 42   0.0000000 |
| FREON-11    0.0000000 | CONSTIT 26   0.0000000 | CONSTIT 43   0.0000000 |
| CONSTIT 10  0.0000000 | CONSTIT 27   0.0000000 | CONSTIT 44   0.0000000 |
| CONSTIT 11  0.0000000 | CONSTIT 28   0.0000000 | CONSTIT 45   0.0000000 |
| CONSTIT 12  0.0000000 | CONSTIT 29   0.0000000 | CONSTIT 46   0.0000000 |
| CONSTIT 13  0.0000000 | CONSTIT 30   0.0000000 | CONSTIT 47   0.0000000 |
| CONSTIT 14  0.0000000 | CONSTIT 31   0.0000000 | CONSTIT 48   0.0000000 |
| CONSTIT 15  0.0000000 | CONSTIT 32   0.0000000 | CONSTIT 49   0.0000000 |
| CONSTIT 16  0.0000000 | CONSTIT 33   0.0000000 | CONSTIT 50   0.0000000 |
| CONSTIT 17  0.0000000 | CONSTIT 34   0.0000000 |
| CONSTIT 18  0.0000000 | CONSTIT 35   0.0000000 |
|_____________________|_____________________|_____________________|
```

10.43 SPLIT—Flow SPLIT Simulation Description

The SPLIT component provides a general flow split for division of the input stream into two streams with the same composition and fluid properties but different flow rates. The mass, thermal, and pressure calculations are identical to those of the MSPLIT routine. The user should refer to section 10.29 for a discussion of these calculations. The SPLIT is limited to two outlet legs where the MSPLIT may have up to seven.



The location format for the SPLIT is as follows:

LO;SPLIT;EQNAME;Fraction1;Fraction2

where,

Fraction1 = Split fraction of incoming flow to be sent to outlet leg 1 (stream 2)

Fraction2 = Split fraction of incoming flow to be sent to outlet leg 2 (stream 3).

Input Parameters

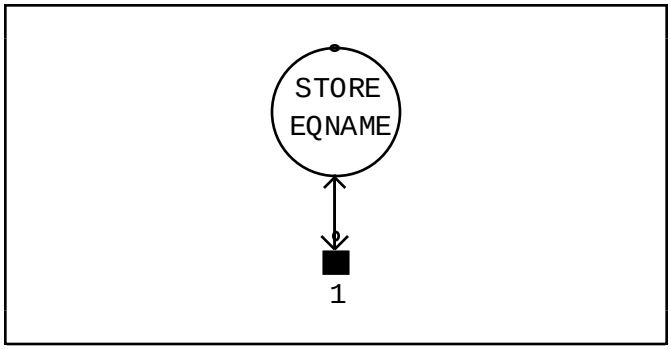
The data base parameters for the SPLIT routine are shown in table 73.

Table 73. SPLIT edit screen.

```
CASE NAME: DEFAULT                                   SPLIT EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:              COMPONENT:                                                                     RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|          ** GENERAL INPUT DATA **          |          ** RESOURCE TRACKING **          |        ** THERMAL INTERFACE DATA **        |
| 10 NUMBER OF OUTLET STREAMS          2.0   | 7 POWER                (WATTS)    0.00   | 17 ** OPEN INPUT LOCATION **       0.000 |
| 13 FLOW SOLUTION DAMPING CONSTNT 0.1000000 | 8 WEIGHT                 (LBM)    0.00   | 18 ** OPEN INPUT LOCATION **       0.000 |
| 11 SPLIT FRACTION OF OUTLT LEG 1 0.0000000 | 9 VOLUME               (FT^3)    0.00   | 19 ** OPEN INPUT LOCATION **       0.000 |
| 12 SPLIT FRACTION OF OUTLT LEG 2 0.0000000 |                                          | 20 ** OPEN INPUT LOCATION **       0.000 |
| 14 HYDRAUL K FACT OF OUTLT LEG 1  0.00E+00 |                                          | 21 ** OPEN INPUT LOCATION **       0.000 |
| 15 HYDRAUL K FACT OF OUTLT LEG 2  0.00E+00 |                                          |                                          |
|____________________________________________|__________________________________________|__________________________________________|
|                      ** BENCHMARK DATA **                                |          ** OUTPUT DATA **                                |
|                          MIN          NOM          MAX                  |        26 OPERATION MODE FLAG            0.000            |
| INLET FLOW    (LBM/HR)     0.000        0.000        0.000              |        23 INLET FLOW     (LBM/HR)        0.000            |
| SPLIT PRESSURE  (PSIA)     0.000        0.000        0.000              |        24 SPLIT PRESSURE   (PSIA)        0.000            |
| FLOW TEMPERATURE   (F)     0.000        0.000        0.000              |        25 FLOW TEMPERATURE    (F)        0.000            |
|_______________________________________________________________________|_________________________________________________________|
```

10.44 STORE—Storage Tank Simulation Description

The STORE simulates a generalized storage tank/accumulator assembly. The thermal balance associated with this component is based on the uniform state—uniform flow energy equation (tank contents) coupled with a standard finite-differerenced tank structure network. Initial total quantities, mass fractions, temperatures, and pressures are required user inputs. Multiple connections to stream 1 are allowed.



10.44.1 General

The routine simulates either a constant pressure or a constant volume reservoir. The constant volume option can be utilized with gases only and the constant pressure option with liquids only.

The constant pressure option is chosen by specifying a positive value for maximum storage volume. The routine then assumes that the maximum storage volume is fixed at this value and that the tank pressure is maintained at the user specified value by a regulation device. The routine calculates the volume of the contents at each time step in the simulation based on the mass and density of the contents. If the maximum storage capacity is exceeded, the excess mass is vented from the tank and a warning message is printed to the screen. The amount of vented mass is tracked during the simulation but the mass is vented to space, thus introducing a mass imbalance on a system level.

The constant volume option is chosen by specifying a negative value for the maximum storage volume. The routine then assumes the magnitude of this entry as the fixed volume of the tank. The user entry for storage pressure is no longer used. The routine calculates the initial storage pressure based on the mass, volume, and temperature of the components.

In the case where the requested flow rate from the tank is greater than the mass available (M < mass flow rate $*\Delta t$) the flow rates are scaled down to equal the mass available. If no mass is available, then the flow rates are set to zero. The logic that scales down the flow rates based on the mass available is performed by each active component whenever they are attached to a tank.

The following section provides a detailed explanation of the mass and energy balance techniques used in modeling this component.

10.44.2 Energy Balance

Calculation of the tank content temperature involves the application of the generalized conservation of energy equation (uniform state–uniform flow) in which the kinetic and potential energy terms are neglected. This equation is expressed as follows:

$$\dot{Q} + \dot{m}_i h_i = \dot{m}_o h_o + \frac{dE}{dt} + \dot{W},$$

where,

E = total internal energy of fluid in the tank, Btu

$\dot{m}_i$ = mass flow rate into tank, lbm/h

$\dot{m}_o$ = mass flow rate out of tank, lbm/h

h_i = $(U+PV)_i$, specific enthalpy of inlet stream, Btu/lbm

h_o = $(U+PV)_o$, specific enthalpy of outgoing stream, Btu/lbm

$\dot{Q}$ = heat transfer rate into system boundary, Btu/h

$\dot{W}$ = control volume work, Btu/h.

It is desirable to manipulate the above equation into a form that will handle both liquids and gases with minimal modification to the basic

equation. Thus, the following simplifying assumptions are made:

A. Mixing of constituents is uniform and instantaneous over the tank volume

B. Specie concentrations and temperatures exit the tank at the calculated ideally mixed conditions

C. Ideal gas behavior (when applicable)

D. Constant pressure and constant volume specific heats are equal for liquids at typical system pressures.

Implementation of the above assumptions into the energy equation yields the following expression for the tank contents temperature:

$$mC_V\frac{dT}{\Delta t} = \dot{m}_i C_{Pi}(T_i - T) + \dot{m}_i(C_{Pi} - C_{Vi})T + \dot{m}_o(C_{Vo} - C_{Po})T + G(T_j - T) + \dot{Q}_H - PdV/778 ,$$

where,

T = tank contents temperature (absolute), ˚R

(mC_V) = instantaneous tank contents capacitance, Btu/˚F

T_i = inlet fluid stream temperature (absolute), ˚R

C_{Pi} = inlet mixture specific heat (constant pressure), Btu/lbm ˚R

C_{Vi} = inlet mixture specific heat (constant volume), Btu/lbm ˚R

$\dot{m}_i$ = inlet total mass flow rate, lbm/h

$\dot{m}_o$ = outlet total mass flow rate, lbm/h

C_{vo} = outlet mixture specific heat (constant volume), same as tank contents C_v, Btu/lbm ˚R

C_{po} = outlet mixture specific heat constant pressure), same as tank contents C_p, Btu/lbm ˚R

T_j = tank structure temperature (absolute), ˚R

G = effective conductance between tank and tank contents, Btu/h ˚F

Q_H = heat transferred into system by heater, Btu/h

V = system volume, ft^3

P = system pressure, psf.

A detailed derivation of the above equation is given in reference 1. Shown in figure 74 is the complete thermal network utilized in the routine. The tank contents equation is finite differenced into an implicit algebraic format and solved by iteration with the other network nodes. The finite differenced form of the contents equation is given as follows:

$$T^{P+1} = \left[\dot{m}_i C_{Pi} T_i^{P+1} + G T_j^{P+1} + \left(\frac{mC_V}{\Delta t}\right)T^P + Q_i - \frac{P(V^{P+1} - V^P)}{778}\right] / \left[\left(\frac{mC_V}{\Delta t}\right) + \dot{m}_o C_{Po} + \dot{m} Y_i C_{Vi} - \dot{m}_o c_{Vo} + G\right] ,$$

where,

T^{P+1} = contents temperature at end of time step, ˚F

T^P = contents temperature at beginning of time step, ˚F

V^{P+1} = contents volume at the end of the time step, ft^3

V^P = contents volume at the beginning of the time step, ft^3

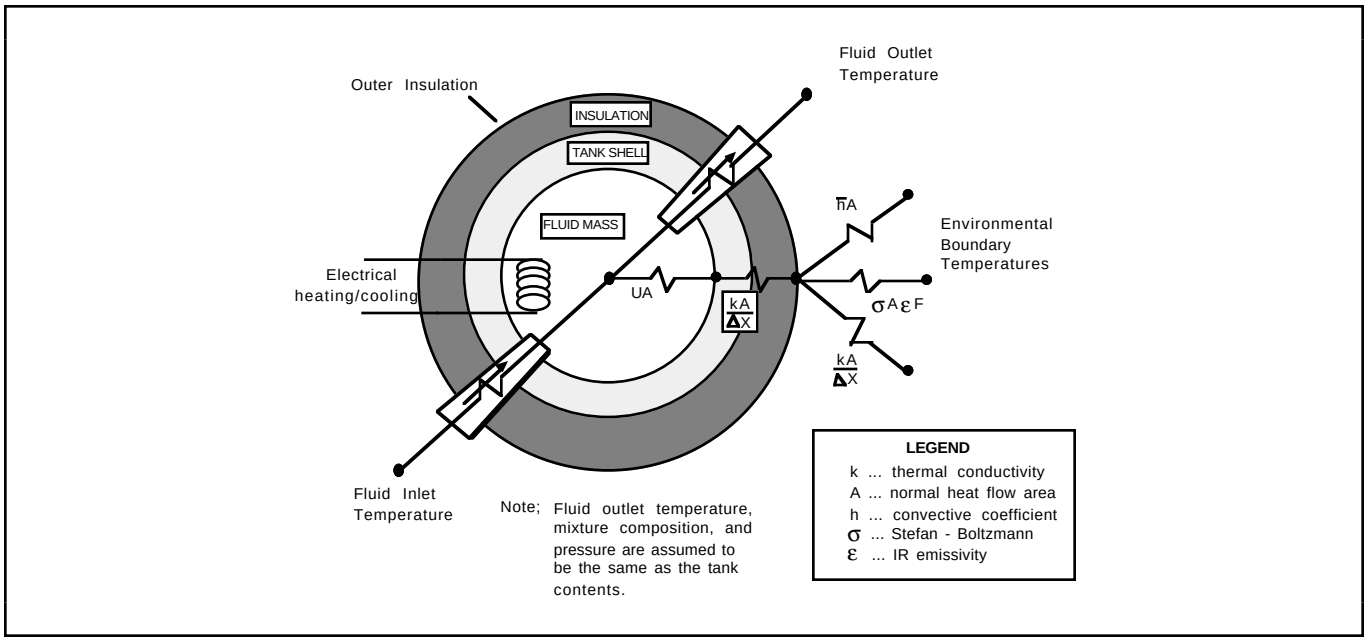


Figure 74. Storage tank thermal network.

10.44.3 Mass Balance

The rate of constituent mass storage is governed by the conservation of mass equation and is given in its finite differenced form (Euler formulation) as:

$$m_i^{P+1} = m_i^P + \Delta t(\sum_i m_i^{in} - \sum_i m_i^{out}),$$

and the total contents mass is given by,

$$m_{total} = \sum_j m_j ,$$

where;

m_i^{P+1} = mass of ith species at the end of time step, lbm

m_i^P = mass of ith species at the beginning of time step, lbm

$m_i^{in/out}$ = mass flow of ith species into (out of) tank, lbm/h

Δt = time increment, h.

Incrementing of the tank contents mass (total and i^{th} constituent) per the explicit formulation given above will occur after the system relaxation criteria has been achieved for the current time step. Numerical integration stability of the Euler formulation is dependent on the time step choice. For the most part, diminishing time steps will give correspondingly greater accuracy to the approximation at the expense of run time. It is therefore left to the user to determine a suitable system time step.

10.44.4 Input Parameters

The input parameters for STORE are presented in table 74. The constituents data base parameters are shown in table 75.

References

1. "G-189 Generalized Environmental/ Thermal Control and Life Support System Analysis Computer Program," 1977.

Table 74. STORE main edit screen.

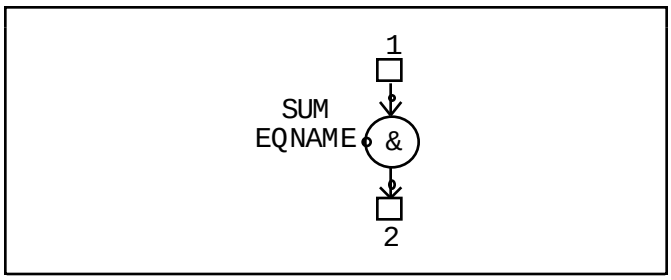
```
CASE NAME: DEFAULT                                  STORE EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                   RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                          ** GENERAL INPUT DATA **                                |          ** THERMAL CHARACTERISTICS **          |
|10 MAX STORAGE VOLUME    (FT^3)   30.0     |69 TEMP SET POINT       (F)    0.00 |                                                 |
|11 STORAGE PRESSURE      (PSIA)     14.70  |70 FLUID VOLUME      (FT^3)    0.00 |12 CONVECTION CONDUCTANCE   (BTU/HR/F)      0.000 |
|17 TANK STRUCTURE MASS    (LBM)    100.00  |      ** RESOURCE TRACKING **       |13 RADIATION CONDUCTANCE  (BTU/HR/R^4)  0.000E+00 |
|                                           |9 RESOURCE VOLUME    (FT^3)    0.00 |14 MOUNTING CONDUCTANCE     (BTU/HR/F)      0.000 |
|18 INITIAL CONTENTS MASS TEMP  (F)   70.00 |8 RESOURCE WEIGHT     (LBM)    0.00 |15 WALL CONDUCTANCE         (BTU/HR/F)      0.000 |
|19 INITIAL CONTENTS MASS (LBM)        0.00 |7 RESOURCE POWER    (WATTS)    0.00 |16 FLUID TO WALL UA         (BTU/HR/F)      0.020 |
|_____________________________________________|_____________________________________|                                                 |
| EMPIRICAL CONSTITUENT DATA (^L TO OPEN) _    (ITEMS 20 TO 68)                     |                                                 |
|_____________________________________________________________________________________|_____________________________________________________|
|                 ** BENCHMARK DATA **                                                |          ** OUTPUT DATA **                          |
|                        MIN         NOM         MAX                                  |                                                 |
| CONTENTS MASS (LBM)   0.00E+00   0.00E+00   0.00E+00|73 FINAL CONTENTS MASS (LBM)    0.00 78 TOTAL ACCUM GAS ADDED (LBM)    0.00 |
| CONTENTS VOL (FT^3)   0.00E+00   0.00E+00   0.00E+00|74 FINAL CONTENTS VOL (FT^3)    0.00 79 TOTL ACCUM GAS VENTED (LBM)    0.00 |
| CONTENTS TEMP   (F)   0.00E+00   0.00E+00   0.00E+00|75 FINAL CONTENTS TEMP   (F)    0.00 80 DELTA ACCUM GAS/STEP  (LBM)    0.00 |
|                                                     |76 VENTED EXCESS MASS  (LBM)    0.00 81 ACCUM GAS SIDE MASS   (LBM)    0.00 |
|                                                     |77 FINAL TANK TEMP       (F)    0.00                                         |
|_____________________________________________________|_____________________________________________________________________________|
```

Table 75. STORE constituents edit screen.

```
CASE NAME: DEFAULT        STORE CONSTITUENT SCREEN        LAST UPDATED: 890215
SUBSYSTEM:               COMPONENT:                      RECORD NUMBER: 000001
_____________________________________________________________________________________________
| CONSTITNT  | INITL MASS | CONSTITNT  | INITL MASS | CONSTITNT  | INITL MASS |
|   LABEL    |  FRACTION  |   LABEL    |  FRACTION  |   LABEL    |  FRACTION  |
|____________|____________|____________|____________|____________|____________|
|CONSTIT 2     0.0000000  |CONSTIT 19    0.0000000  |CONSTIT 36    0.0000000  |
|CONSTIT 3     0.0000000  |CONSTIT 20    0.0000000  |CONSTIT 37    0.0000000  |
|CONSTIT 4     0.0000000  |CONSTIT 21    0.0000000  |CONSTIT 38    0.0000000  |
|CONSTIT 5     0.0000000  |CONSTIT 22    0.0000000  |CONSTIT 39    0.0000000  |
|CONSTIT 6     0.0000000  |CONSTIT 23    0.0000000  |CONSTIT 40    0.0000000  |
|CONSTIT 7     0.0000000  |CONSTIT 24    0.0000000  |CONSTIT 41    0.0000000  |
|CONSTIT 8     0.0000000  |CONSTIT 25    0.0000000  |CONSTIT 42    0.0000000  |
|CONSTIT 9     0.0000000  |CONSTIT 26    0.0000000  |CONSTIT 43    0.0000000  |
|CONSTIT 10    0.0000000  |CONSTIT 27    0.0000000  |CONSTIT 44    0.0000000  |
|CONSTIT 11    0.0000000  |CONSTIT 28    0.0000000  |CONSTIT 45    0.0000000  |
|CONSTIT 12    0.0000000  |CONSTIT 29    0.0000000  |CONSTIT 46    0.0000000  |
|CONSTIT 13    0.0000000  |CONSTIT 30    0.0000000  |CONSTIT 47    0.0000000  |
|CONSTIT 14    0.0000000  |CONSTIT 31    0.0000000  |CONSTIT 48    0.0000000  |
|CONSTIT 15    0.0000000  |CONSTIT 32    0.0000000  |CONSTIT 49    0.0000000  |
|CONSTIT 16    0.0000000  |CONSTIT 33    0.0000000  |CONSTIT 50    0.0000000  |
|CONSTIT 17    0.0000000  |CONSTIT 34    0.0000000  |                         |
|CONSTIT 18    0.0000000  |CONSTIT 35    0.0000000  |                         |
|____________________________|____________________________|____________________________|
```

10.45 SUM—Flow Summation Simulation Description

The SUM routine provides a flow stream combination function. Any number of streams may be combined into a single outlet stream. The composition and flow rate of the outlet stream are calculated under the assumption that no chemical reactions occur.



10.45.1 Thermal Calculations

The temperature of the outlet stream is calculated as the weighted average of the temperatures of the incoming streams:

$$T_o = \frac{\sum_{j=1}^{n} \dot{m}_j C_{p_j} T_j}{\sum_{j=1}^{n} \dot{m}_j C_{p_j}},$$

where,

T_o = outlet stream temperature (°F)

T_j = inlet stream j temperature (°F)

$\dot{m}_j$ = inlet stream j mass flow (lbm/h)

C_{p_j} = inlet stream j specific heat (Btu/lbm/°F)

n = number of inlet streams.

10.45.2 Pressure Calculations

The pressure of the SUM is calculated according to the following equation:

$$P_S = \frac{\sum_{j=1}^{k} G_j P_j}{\sum_{j=1}^{k} G_j},$$

where,

P_s = pressure of the SUM (lb/in^2 absolute)

P_j = pressure of connecting stream j (lb/in^2 absolute)

G_j = flow conductance of connection j (lbm/h/lb/in^2 absolute)

k = total number of streams connected to the SUM.

The flow conductance of each connection is given by the quasi-linear approximation:

$$G_j = \frac{\dot{m}_j}{\Delta P_j},$$

where,

ΔP_j = the pressure drop for connection j (lb/in^2 absolute)

The connection pressure drops are calculated by the solution system routines discussed in section 8.2.

10.45.3 Input Parameters

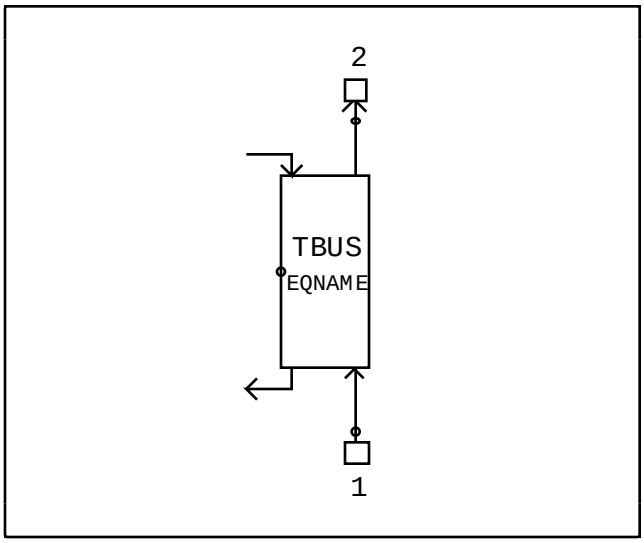
The data base parameters for the SUM are shown in table 76.

Table 76. SUM edit screen.

```
CASE NAME: DEFAULT                                      SUM EDIT SCREEN                                      LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                        RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                                                   ** GENERAL INPUT DATA **                                                  |
| 10 FLOW DAMPING FACTOR          0.100     | 20 INPUT CONSTANT #11          0.000E+00 | 30 INPUT CONSTANT #21          0.000E+00 |
| 11 INPUT CONSTANT  #2           0.000E+00 | 21 INPUT CONSTANT #12          0.000E+00 | 31 INPUT CONSTANT #22          0.000E+00 |
| 12 INPUT CONSTANT  #3           0.000E+00 | 22 INPUT CONSTANT #13          0.000E+00 | 32 INPUT CONSTANT #23          0.000E+00 |
| 13 INPUT CONSTANT  #4           0.000E+00 | 23 INPUT CONSTANT #14          0.000E+00 | 33 INPUT CONSTANT #24          0.000E+00 |
| 14 INPUT CONSTANT  #5           0.000E+00 | 24 INPUT CONSTANT #15          0.000E+00 | 34 INPUT CONSTANT #25          0.000E+00 |
| 15 INPUT CONSTANT  #6           0.000E+00 | 25 INPUT CONSTANT #16          0.000E+00 | 35 INPUT CONSTANT #26          0.000E+00 |
| 16 INPUT CONSTANT  #7           0.000E+00 | 26 INPUT CONSTANT #17          0.000E+00 | 36 INPUT CONSTANT #27          0.000E+00 |
| 17 INPUT CONSTANT  #8           0.000E+00 | 27 INPUT CONSTANT #18          0.000E+00 | 37 INPUT CONSTANT #28          0.000E+00 |
| 18 INPUT CONSTANT  #9           0.000E+00 | 28 INPUT CONSTANT #19          0.000E+00 | 38 INPUT CONSTANT #29          0.000E+00 |
| 19 INPUT CONSTANT #10           0.000E+00 | 29 INPUT CONSTANT #20          0.000E+00 | 39 INPUT CONSTANT #30          0.000E+00 |
|                                           |          ** RESOURCE TRACKING **          |                                          |
|  7 POWER                  (WATTS)    0.00 |  8 WEIGHT                  (LBM)    0.00 |  9 VOLUME                 (FT^3)    0.00 |
|_____________________________________________|_____________________________________________|_____________________________________________|
|                    ** BENCHMARK DATA **                   |                          ** OUTPUT DATA **                                |
|                                                           |                                                                           |
| BENCHMARK   #1   0.00E+00   #2   0.00E+00   #3   0.00E+00 | 40 OUTPUT CONSTANT #1        0.00   43 OUTPUT CONSTANT #4      0.00  |
| BENCHMARK   #4   0.00E+00   #5   0.00E+00   #6   0.00E+00 | 41 OUTPUT CONSTANT #2        0.00   44 OUTPUT CONSTANT #5      0.00  |
| BENCHMARK   #7   0.00E+00   #8   0.00E+00   #9   0.00E+00 | 42 OUTPUT CONSTANT #3        0.00   45 OUTPUT CONSTANT #6      0.00  |
|_____________________________________________________________________________________________________________________________|
```

10.46 TBUS—Thermal Bus Simulation Description

This routine simulates the thermal control behavior exhibited by a thermal bus (TBUS). The simplified algorithm used to determine the amount of heat transferred is based on an isothermal control fluid inlet temperature. The TBUS does not generate flow; therefore, an active component must be upstream of the inlet. Pressure drop through the TBUS is determined by a user-supplied equivalent line length and equivalent line diameter. The TBUS data base parameters are shown in table 77.



Discussion

The TBUS simulation assumes an isothermal control fluid temperature such as would be encountered with a two-phase fluid or a fluid with a large heat capacity rate. The equation used for calculating the outlet stream temperature utilizes the delta network approach described in section 8.3.2. The TBUS utilizes a thermal network equivalent to the CNHX network (see fig. 60) that requires the user to supply a convective effectivity. A delta network solution scheme is used to solve for the outlet conditions of the working fluid. The effectivity can be a constant or can be calculated dynamically in the OPS code (if changing operating conditions warrant it) with an equation such as the following:

$$\varepsilon = 1 - \exp(hA/\dot{m}C_p) ,$$

where,

h = internal heat transfer coefficient, Btu/h/ft^2/°F

A = heat exchange area, ft^2

$\dot{m}Cp$ = fluid heat capacity rate, Btu/h/°F.

The internal heat transfer coefficient (h) may be calculated in OPS code using conventional convection correlations for internal flow.

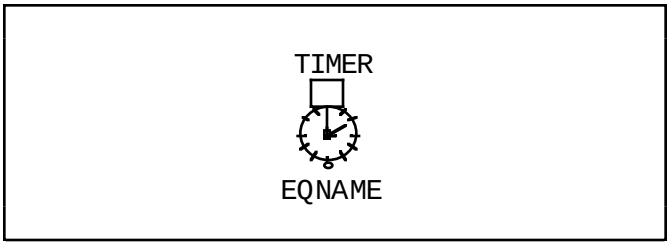
Pressure drop through the TBUS is found by conventional pressure drop calculation methods utilizing a user-supplied equivalent length and equivalent diameter. See section 8.2.5 for a detailed explanation of the pressure drop calculations.

Table 77. TBUS edit screen.

```
CASE NAME: DEFAULT              TBUS EDIT SCREEN              LAST UPDATED: 890215
SUBSYSTEM:                     COMPONENT:                     RECORD NUMBER: 000001
_____________________________________________________________________________________
|            ** GENERAL INPUT DATA **                |                                |
|                                                    |   ** RESOURCE TRACKING **      |
|10 EFFECTIVITY                            0.90000   |7 POWER    (WATTS)    0.00      |
|11 CONTROL FLUID TEMPERATURE        (F)    70.00   |8 WEIGHT     (LBM)    0.00      |
|                                                    |9 VOLUME    (FT^3)   0.000      |
|         ** PRESSURE DROP PARAMETERS **             |                                |
|12 EQUIVALENT LENGTH                (FT)    10.00   |                                |
|13 EQUIVALENT DIAMETER              (IN)    0.500   |                                |
|____________________________________________________|________________________________|
|              ** BENCHMARK DATA **                  |     ** OUTPUT DATA **          |
|                     MIN        NOM        MAX      |                                |
| OUTLET TEMP  (F)  0.00E+00   0.00E+00   0.00E+00   |14 HEAT TRNSFR (W)     0.00     |
| HEAT TRANSFR (W)  0.00E+00   0.00E+00   0.00E+00   |                                |
|____________________________________________________|________________________________|
```

10.47 TIMER—Timer Simulation Description

The TIMER component provides the capability to timeline multiple parameters in any component's performance data. The user may specify either a list of discrete timed events relative to the system simulation time or a list of events to be repeated on a specified cycle period. The TIMER icon is shown below.



10.47.1 General Discussion

The TIMER is placed on the subsystem screen with the component that is to be controlled. A connection is then established between the TIMER hit box and the controlled component's body hit dot with the connect command. The interconnecting control line is drawn as a dotted line to distinguish it from a normal fluid connection line. The TIMER must be connected only to the component body hit dot and not to a fluid stream hit box. This tells the CASE/A system that component is associated with the TIMER's control actions. More than one TIMER may be connected to the same component. A TIMER may itself be controlled by another TIMER if the user wishes to nest them.

10.47.2 User Inputs

The TIMER's basic operation is controlled by three user inputs. It may be turned on and off by setting the operation mode (input 10) to 1.0 or 0.0, respectively (see table 78). If the operation mode is set to "on," the TIMER logic is not executed until the activation time (input 11) is reached. The default activation time is set to 0.0, which means it will be active at the beginning of the simulation. The TIMER logic is skipped after the deactivation time (input 12) is reached. The default deactivation time is 99,999 h.

The TIMER allows the user to specify two different control schemes; discreet timed "events" and repetitive cyclic "events." The scheme is set by the TIMER cycle duration (input 13). If the cycle duration is set to a value of zero, the TIMER operates in the discreet event mode. In this mode, the list of timed events is processed relative to the simulation elapsed time. If the cycle duration is set to a nonzero value, the TIMER operates in the cyclic event mode. In this mode, the list of events is repeated based on the cycle duration: an event's activation time is relative to the start of each cycle. The default cycle duration is zero (discrete events).

The TIMER allows multiple component constants to be controlled. The input format consists of three items per each individual timed event for either control mode:

(1) Parameter location number

(2) Event activation time

(3) Parameter value.

The parameter location number is the relative number in the data base of the component parameter to be controlled. These numbers are displayed on the edit screens of each component to the left of each individual parameter (see sections 3 and 10).

The activation time of the event is the time at which the component parameter is to be set. The user must realize that this time depends on the control mode of the TIMER: discreet events or cyclic repetition as discussed above.

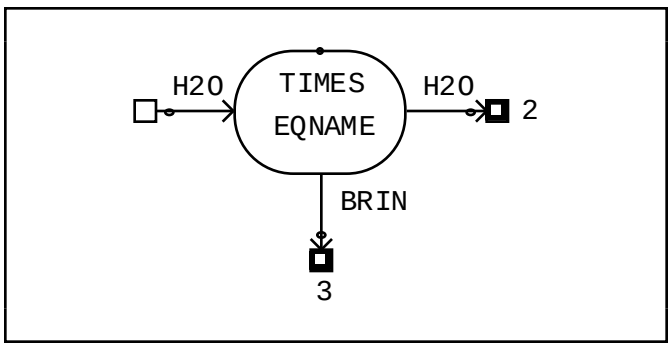
The parameter values may be set according to the desires of the user. They will be left at the last values set by the user upon deactivation of the TIMER or simulation termination.

Table 78. TIMER edit screen.

```
CASE NAME: TIMER              TIMER EDIT SCREEN            LAST UPDATED: 000000
SUBSYSTEM: DEFAULT            COMPONENT: VALUES           RECORD NUMBER: 000001
_____________________________________________________________________________
|                           ** GENERAL INPUT DATA **                        |
| 10 OPERATION MODE (0=OFF, 1=ON)   1. | 13 CYCLE DURATION, Hr          0.0000 |
| 11 ACTIVATION   TIME,Hr       0.0000 | 14 *** OPEN INPUT ***          0.0000 |
| 12 DEACTIVATION TIME,Hr   99999.0000 | 15 *** OPEN INPUT ***          0.0000 |
|_____________________________________________________________________________|
|                           ** TIMED EVENT INPUTS **                        |
|EVT  LOC  ACTVTN TIM,Hr     VALUE       EVT  LOC  ACTVTN TIM,Hr    VALUE     |
|  1    0.        0.0000       0.000000 | 14    0.      0.0000       0.000000  |
|  2    0.        0.0000       0.000000 | 15    0.      0.0000       0.000000  |
|  3    0.        0.0000       0.000000 | 16    0.      0.0000       0.000000  |
|  4    0.        0.0000       0.000000 | 17    0.      0.0000       0.000000  |
|  5    0.        0.0000       0.000000 | 18    0.      0.0000       0.000000  |
|  6    0.        0.0000       0.000000 | 19    0.      0.0000       0.000000  |
|  7    0.        0.0000       0.000000 | 20    0.      0.0000       0.000000  |
|  8    0.        0.0000       0.000000 | 21    0.      0.0000       0.000000  |
|  9    0.        0.0000       0.000000 | 22    0.      0.0000       0.000000  |
| 10    0.        0.0000       0.000000 | 23    0.      0.0000       0.000000  |
| 11    0.        0.0000       0.000000 | 24    0.      0.0000       0.000000  |
| 12    0.        0.0000       0.000000 | 25    0.      0.0000       0.000000  |
| 13    0.        0.0000       0.000000 |                                      |
|_____________________________________________________________________________|
```

10.48 TIMES—Thermoelectric Integrated Membrane Evaporation Subsystem Simulation Description

This routine simulates the performance of a thermoelectric integrated membrane evaporation system (TIMES). It allows for calculation of product water quality, product water quantity, and power consumption. The principal user-specified performance parameters include constituent separation efficiencies, maximum allowable percent solids in the feed, average specific energy required, and water production rate versus percent solids characteristics. The routine allows for heat exchange between the component and its surroundings. The TIMES does not generate flow; therefore, an active component must be upstream of the inlet. The TIMES primary data base parameters are shown in table 79. The TIMES constituent data base parameters are shown in table 80.



Discussion

The TIMES water recovery characteristics are simulated using the percent solid in feed versus water production rate empirical data available in the literature. The feed rate is divided into the water production rate to obtain the fraction of the water in the feed that is distilled. The resulting relationship of percent solids versus the distilled fraction of water in the feed is linearized as shown in figure 75. The slope and Y-intercept of this line can be entered by the user through normal component parameter editing methods.

The carry-over of contaminants into the distillate stream is determined from the separation efficiencies input by the user for each constituent. The separation efficiency fraction for each constituent is given by the following equation:

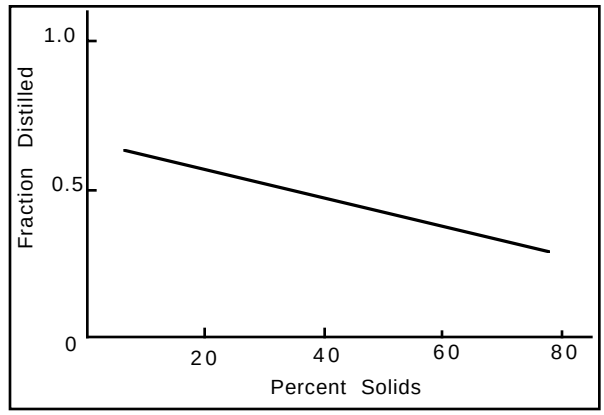


Figure 75. Percent solids versus fraction of water in feed distilled (example only).

$$E_s = 1 - \frac{\dot{m}_d X_d}{\dot{m}_i X_i} ,$$

where,

E_s = separation efficiency for individual constituents

$\dot{m}_d$ = TIMES distillate stream total flow rate, lbm/h

$\dot{m}_i$ = TIMES inlet waste total flow rate, lbm/h

X_d = constituent concentration in distillate stream, ppm

X_i = constituent concentration in inlet waste stream, ppm.

The maximum allowable percent solids in the feed can also be user defined. In the case under consideration, if the percent solids in the feed reaches the maximum specified by the user, then all of the feed is diverted to the brine stream (no distillate is produced). A message appears to the screen notifying the user of the condition.

Another user input is the specific energy requirements for the distillation process. The TIMES subroutine multiplies the specific energy by the water production rate at a given time to yield the power consumption. Nominal power consumption is calculated by multiplying specific energy by the cumulative water produced and dividing by the total elapsed time of distillate production. If the percent solids is above the maximum specified (resulting in a bypass

condition) then the power consumption is considered to be zero.

This routine tracks stream temperature and pressure, and allows for heat exchange between the TIMES and its surroundings. A delta network of one-way conductors is used to simulate the thermal behavior of the TIMES. This method requires the user to input an effectivity for the network. The TIMES is considered to exchange heat with its surroundings through convective, conductive, and radiative heat transfer mechanisms (see fig. 76). The convective, conductive, and radiative heat transfer coefficients can be input by the user. The previously mentioned effectivity allows the user to control the degree to which the TIMES fluid streams are influenced by the power consumption and the heat exchange with the ambient. It is assumed that the product water stream and the brine stream exit the TIMES at the same temperature.

Since the TIMES is sensitive to the temperature at which it processes water, the user has the capability to set a maximum exit fluid temperature. If the calculated outlet fluid temperature is above the maximum desired, then the additional energy of the outlet streams is diverted to the cabin through convective pathways. This is done to simulate the ambient/condensate HX integral to the TIMES in its present design configuration. The product stream temperatures are then set to the user-supplied maximum.

The TIMES is considered to have a set outlet pressure for the brine and product water streams. This is considered to be a reasonable assumption since the TIMES has integral fluid pumps that dictate the pressure of the outlet streams.

Table 79. TIMES edit screen.

```
CASE NAME: DEFAULT          TIMES EDIT SCREEN              LAST UPDATED: 890805
SUBSYSTEM:                COMPONENT:                     RECORD NUMBER: 000001
_____________________________________________________________________________
|          ** GENERAL INPUT DATA **          |   ** THERMAL INTERFACE DATA **  |
| 10 % SOLIDS SHUTDOWN POINT         38.00   |                                 |
| 11 SLOPE %SOL-FRAC H20 PROD LINE -0.0001   | 15 TIM-ENV H-A (BTU/HR/F)   0.000 |
| 12 Y-INT %SOL-FRAC H20 PROD LINE  0.0060   | 16 TIME-ENV F-A-E  (FT^2)   0.000 |
|                                            | 17 TIM-EN KA/X (BTU/HR/F)   0.000 |
| 13 SPEC ENERGY CONS   (W-HR/LBM)   85.00   | 18 FLUID-WAL Q TRAN EF <1 0.90000 |
|                                            | 19 MAX FLUID OUT TEMP (F)  155.00 |
| 14 INITIAL TIMES TEMPERATURE (F)   70.00   |     ** RESOURCE TRACKING **       |
| 20 PRODUCT STREAM PRESSUR (PSIA)   20.00   | 7 POWER           (WATTS)    0.00 |
| 21 BRINE STREAM PRESSURE  (PSIA)   18.00   | 8 WEIGHT            (LBM)    0.00 |
|                                            | 9 VOLUME           (FT^3)    0.00 |
| CONSTITUENT EMPIRICAL DATA (^L TO OPEN)_   |                                 |
|____________________________________________|_________________________________|
|                         ** BENCHMARK AND OUTPUT DATA **                     |
|                            MIN        NOM        MAX                        |
| WASTE IN FLOW  (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                     |
| BRINE OUT FLOW (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                     |
| PROD OUT FLOW  (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                     |
| POWER CONSUMED  (WATTS)  0.00E+00   0.00E+00   0.00E+00                     |
| 75 TIMES TEMP       (F)    0.00                                             |
|_____________________________________________________________________________|
```

Table 80. TIMES constituent edit screen.

```
CASE NAME: DEFAULT       TIMES CONSTITUENT SCREEN         LAST UPDATED: 890805
SUBSYSTEM:               COMPONENT:                     RECORD NUMBER: 000001
_____________________________________________________________________________
|    LABEL    |  SEP EFFCY  |    LABEL    |  SEP EFFCY  |    LABEL    |  SEP EFFCY  |
|             |  FRACTION   |             |  FRACTION   |             |  FRACTION   |
|_____________|_____________|_____________|_____________|_____________|_____________|
| CONSTIT  2  |  1.0000000  | CONSTIT 19  |  1.0000000  | CONSTIT 36  |  1.0000000  |
| CONSTIT  3  |  1.0000000  | CONSTIT 20  |  1.0000000  | CONSTIT 37  |  1.0000000  |
| CONSTIT  4  |  1.0000000  | CONSTIT 21  |  1.0000000  | CONSTIT 38  |  1.0000000  |
| CONSTIT  5  |  1.0000000  | CONSTIT 22  |  1.0000000  | CONSTIT 39  |  1.0000000  |
| CONSTIT  6  |  1.0000000  | CONSTIT 23  |  1.0000000  | CONSTIT 40  |  1.0000000  |
| CONSTIT  7  |  1.0000000  | CONSTIT 24  |  1.0000000  | CONSTIT 41  |  1.0000000  |
| CONSTIT  8  |  0.0000000  | CONSTIT 25  |  1.0000000  | CONSTIT 42  |  1.0000000  |
| CONSTIT  9  |  1.0000000  | CONSTIT 26  |  1.0000000  | CONSTIT 43  |  1.0000000  |
| CONSTIT 10  |  1.0000000  | CONSTIT 27  |  1.0000000  | CONSTIT 44  |  1.0000000  |
| CONSTIT 11  |  1.0000000  | CONSTIT 28  |  1.0000000  | CONSTIT 45  |  1.0000000  |
| CONSTIT 12  |  1.0000000  | CONSTIT 29  |  1.0000000  | CONSTIT 46  |  1.0000000  |
| CONSTIT 13  |  1.0000000  | CONSTIT 30  |  1.0000000  | CONSTIT 47  |  1.0000000  |
| CONSTIT 14  |  1.0000000  | CONSTIT 31  |  1.0000000  | CONSTIT 48  |  1.0000000  |
| CONSTIT 15  |  1.0000000  | CONSTIT 32  |  1.0000000  | CONSTIT 49  |  1.0000000  |
| CONSTIT 16  |  1.0000000  | CONSTIT 33  |  1.0000000  | CONSTIT 50  |  1.0000000  |
| CONSTIT 17  |  1.0000000  | CONSTIT 34  |  1.0000000  |             |             |
| CONSTIT 18  |  1.0000000  | CONSTIT 35  |  1.0000000  |             |             |
|_____________|_____________|_____________|_____________|_____________|_____________|
```

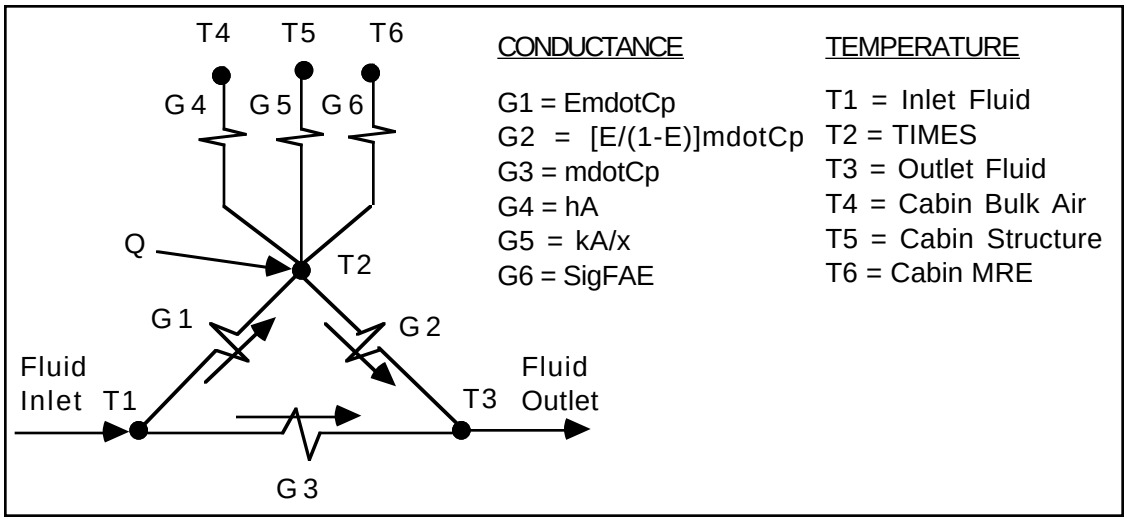
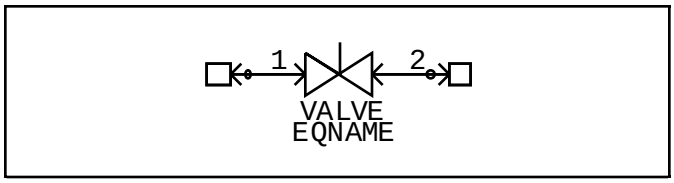


Figure 76. TIMES heat transfer network.

10.49 VALVE—VALVE Simulation Description

The VALVE component routine simulates the pressure drop and flow control characteristics of a variety of valve types. The component allows either one-way or bi-directional flow specification to simulate check valves or normal valves. It also offers limited simulation of compressible flow through square edged orifices (does not support choked flow). The VALVE component is a "passive" device with respect to mass flow—it only accepts the upstream mass flow or relies on a downstream component to pull flow through it. Therefore, an "active" flow-generating component such as a PUMP or NODE must be located somewhere upstream or downstream of the VALVE. The icon configuration is shown below.



10.49.1 General Assumptions

The solution of the pressure drop for the normal valve types is based upon Darcy's formula that is applicable to all fluids. This formula provides a means of analyzing flow through valves on the basis of equivalent length and equivalent diameter. The orifice thermal calculations are based on the assumption of reversible adiabatic expansion through the orifice and the application of ideal gas relations.

10.49.2 Mass Calculations

The initial valve position fraction is set by the user via an input parameter in the data base. This initial position value, which ranges from a fully closed value of 0.0 to a fully open value of 1.0, is copied into an output parameter location that is used inside the routine to indicate the valve position at any time thereafter. Therefore, when the user wishes to control the valve position during a simulation via OPS logic or controller action, this output parameter must be varied instead of the initial value in the input location. This dual location setup is implemented to provide flexibility for the user. Further discussion of the control of the valve flow characteristics is contained in the pressure drop section of this write-up.

The direction of the fluid flow may be specified by the user as being one of three possible choices:

A. Only from stream 1 to stream 2

B. Only from stream 2 to stream 1

C. Conditionally determined by the routine itself depending on the connecting component stream pressures.

The flow direction flag is set by the user as an input parameter in the VALVE data base. If the user sets the direction incorrectly for a given flow network, such as setting the flow direction opposite to the flow from the outlet of a PUMP, the solution system will recognize this condition and print a warning message to the screen. However, the valve will act as an ideal check valve with zero cracking pressure differential when the flow direction is set and a flow balancing component such as the NODE attempts to "draw" or "push" flow in the opposite direction. The user should always set the correct flow direction if it is known in order to speed the convergence of the overall system flow and pressure balance solutions, especially when NODE components or other flow balancing components are involved.

The fluid composition is merely passed from the functional inlet stream of the VALVE straight through to the outlet stream with no leakage loss or composition change.

10.49.3 Thermal Calculations

An R-C network analysis technique is utilized in the thermal calculations and is based on the generic network discussed in section 8.3.3 that allows for three modes of heat transfer with the environment: convection, conduction, and radiation. The network solution calculates the fluid outlet temperature and structural temperatures based on the user specified conductances, thermal capacitance of the mass node, and environment and fluid inlet boundary temperatures. There are four linear conductances the user must provide as inputs along with the emissive area term for radiation and a fluid-to-wall heat transfer effectivity. Also, the user must

specify an initial mass temperature, a relaxation criterion for all temperatures, and a maximum iteration count not to be exceeded in the solution routine. The user should refer to the discussion in section 8.3.3 for detailed information on the network and the solution method involved. However, the VALVE routine does differ from the standard thermal network in that an expansion of a compressible fluid results in a temperature change of the fluid. Therefore, the following methodology is used in the thermal calculations for the flow configurations involving expansion across the valve:

Step 1:

The fluid temperature change due to expansion is calculated according to:

$$To = Ti\,(Po/Pi)^{(k-1)/k} \ ,$$

where,

Ti = the fluid inlet temperature (°R)

To = the fluid temperature after expansion (°R)

Pi = the fluid inlet pressure (lb/in^2 absolute)

Po = the fluid outlet pressure (lb/in^2 absolute)

k = the fluid specific heat ratio = Cp/Cv .

Note that this formula applies to a perfect gas with constant specific heats undergoing a reversible adiabatic process. There is no temperature change due to expansion for an incompressible fluid.

Step 2:

This expansion temperature is then used as the fluid inlet boundary temperature in the transient thermal network solution discussed above. The valve fluid outlet and structural temperatures are then set to the values returned by the network solution routine.

This methodology is not rigorously correct since the actual process falls into the category of Rayleigh flow but it is assumed that sufficient accuracy will be obtained with this approach for most cases.

10.49.4 Pressure Calculations

The VALVE routine provides three different methods of pressure drop versus flow rate calculations: the equivalent length method; the resistance coefficient method; and orifice flow calculations. The routine calculates the sum of all three methods and uses this value as the total pressure drop across the valve. The user should set the appropriate input constants in order to employ the correct method(s). The three methods and their controlling strategies are discussed below.

10.49.5 Equivalent Length Method

The equivalent length method stems from the Darcy formula for internal frictional pressure loss:

$$(Pi-Po) = \frac{2\rho f L(v)^2}{144\,(D\,g)} \ ,$$

where,

Pi = the inlet pressure (lb/in^2 absolute)

Po = the outlet pressure (lb/in^2 absolute)

ρ = the fluid density (lbm/ft^3)

L = the equivalent length (ft) (user input)

v = the fluid velocity (ft/s)

D = the equivalent diameter (in) (user input)

g = Newton's constant (32.17 ft*lbm/lbf/s/s)

f = the Fanning friction factor (function of Reynolds number).

The VALVE routine calls the generic solution routine FRICTDP that calculates the pressure drop for the equivalent length method and is discussed in section 8.2.4 in greater detail. The valve position parameter offers no control over the equivalent length of the valve. Therefore, this method essentially only allows the simulation of a fully open or closed valve unless the user dynamically sets the length from within the OPS logic blocks or through the actions of a CNTRLLR component. If the user wishes to control the valve in this manner, the valve position should be set in the fully open position (X = 1.0) and the equivalent length set at the user

determined value. The valve may be shut off by setting the position fraction to zero (X = 0.0). The length should be set to a value of zero to disable the equivalent length method.

10.49.6 Resistance Coefficient Method

The resistance coefficient method uses the following equation to characterize the pressure drop across the valve:

$$(Pi–Po) = C\,(\dot{m} / X)^2 ,$$

where,

Pi = the inlet pressure (lb/in^2 absolute)

Po = the outlet pressure (lb/in^2 absolute)

C = the pressure drop factor (empirical user input)

X = the valve position fraction (0.0 = closed, 1.0 = open)

$\dot{m}$ = the mass flow rate (lbm/h).

The valve position offers direct control over the pressure loss of the valve. The user may wish to simulate various types of valve opening characteristics (i.e., linear, equal percentage, or quick opening) by appropriate manipulation of the valve position and/or the pressure drop factor through OPS logic or CNTRLLR actions. The pressure drop factor should be set to a value of zero to disable this option.

10.49.7 Orifice Flow Calculations

The orifice option is limited to the calculation of the pressure drop for the existing inlet pressure and mass flow entering the valve. It may be used for liquid flows as well as subsonic gas flows but not for choked flow conditions. There is no check for choked flow occurring across the orifice. The flow through an orifice is given by the empirical formula (ref 1):

$$\dot{m} = KA_oY\sqrt{2g\rho(Pi–Po)144} ,$$

where,

$\dot{m}$ = the fluid flow rate (lbm/h)

A_o = the orifice cross-sectional area (ft^2)

Pi = the orifice inlet pressure (lb/in^2 absolute)

Po = the orifice outlet pressure (lb/in^2 absolute)

ρ = the density of the fluid (lbm / ft^3)

K = the orifice empirical geometry coefficient

Y = the orifice compressibility correction factor.

Note the formulas for A_o, K, and Y are given by:

$$A_o = X\,Am$$

$$K = \frac{Cd}{\sqrt{1.0–\beta^4}},$$

$$Y = 1.0–(0.41+0.35\beta^4)\,\frac{(Pi–Po)}{kPi} ,$$

$$\beta = A_o/A_i ,$$

where,

Am = the area at full open position (user input, ft^2)

Ai = the upstream pipe cross-sectional area (ft^2)

Cd = the discharge coefficient (user input, $0 < Cd <= 1.0$)

β = the area ratio (user input, $0 < \beta <= 1.0$)

k = the specific heat ratio = (Cp/Cv)

X = the valve position fraction (0.0 = closed, 1.0 = open).

An implicit algorithm is utilized wherein back substitution of the compressibility factor “Y” and the orifice outlet pressure “Po” is carried out until convergence is obtained. The compressibility factor is set to 1.0 for incompressible fluids (k = 1.0). The valve position offers direct control over

the orifice area according to the formula for *Ao* shown above. Note that the full open value for the area ratio should be entered in the user input section of the data base. The full open area should be set to zero to disable this option.

10.49.8 Input Parameters

The VALVE data base parameters are shown in table 81. The default data are set such that the valve is initially fully open, there is no pressure drop, the flow direction is from stream 1 to stream 2, no heat exchange to the environment occurs and the valve has no thermal capacitance (steady-state thermal solution).

Reference

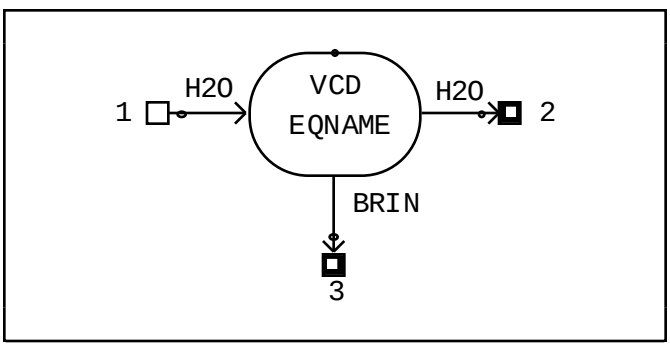
1. "ASHRAE Handbook 1981 Fundamentals," American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., 1981.

Table 81. VALVE edit screen.

```
CASE NAME: DEFAULT                                   VALVE EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:              COMPONENT:                                                                    RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|                ** INPUT DATA **                |    ** THERMAL CHARACTERISTICS DATA **     |     ** MISCELLANEOUS  PARAMETERS **     |
| 10 INITIAL POSITION   (0<=X<=1) 1.0000000      | 18 SHELL-ENV CONVCT (BTU/HR/F)     2.000  | 28  ACTIVE FLOW,  0=NO  1=YES     0.000 |
| 11 EQUIVALENT LENGTH       (FT)     0.000      | 19 SHELL-ENV RAD FAE    (FT^2)     1.000  | 29  ** OPEN INPUT LOCATION **     0.000 |
| 12 EQUIVALENT DIAMETER     (IN)     0.000      | 20 SHELL-ENV CONDCT (BTU/HR/F)     2.000  |                                         |
| 13 K FACT (PDROP=K*(FLOW/X)**2) 0.000E+00      | 21 MASS-SHEL CONDCT (BTU/HR/F)     0.500  |                                         |
| 14 FULL OPEN FLOW X-AREA  (FT2)     0.000      | 22 WALL-MASS CONDCT (BTU/HR/F)     1.000  |          ** RESOURCE TRACKING **        |
| 15 VALVE DISCHGE COEFF (0<Cd<1) 0.6200000      | 23 FLUID-WALL EFFECT   (0<E<1) 0.9500000  | 7  POWER              (WATTS)     0.000 |
| 16 ORIFCE RATIO,Aor/Ain (0<B<1) 0.7500000      | 24 THERMAL CAPACITNCE  (BTU/F)     0.000  | 8  WEIGHT               (LBM)     0.000 |
| 17 DIRECT FLAG(0=?,1=1-2,2=2-1) 1.0000000      | 25 INITIAL MASS TEMPRTURE  (F)    75.000  | 9  VOLUME              (FT^3)     0.000 |
|                                                | 26 THERMAL RELAXATN CRITERIA   0.0000100  |                                         |
|                                                | 27 THERMAL MAX ITERATN COUNT     100.000  |                                         |
|_______________________________________________|_____________________________________|_________________________________________|
|                    ** BENCHMARK DATA **                    |                    ** OUTPUT DATA **                                |
|                          MIN          NOM          MAX     |  30  PRESENT VALVE POSITION            (FRACT) 0.0000000          |
| VALVE POSITION      (FRACT) 0.0000000   0.0000000   0.0000000 |  31  PRESENT PRESSURE DROP             (PSID)     0.000          |
| PRESSURE DROP        (PSID)     0.000       0.000       0.000 |  32  PRESENT MASS FLOW RATE           (LBM/HR)     0.000          |
| MASS FLOW RATE     (LBM/HR)     0.000       0.000       0.000 |  33  PRESENT MASS TEMPERATURE             (F)     0.000          |
|                                                                |  34  PRESENT SHELL TEMPERATURE            (F)     0.000          |
|                                                                |  35   **  OPEN OUTPUT LOCATION  **                0.000          |
|________________________________________________________________|___________________________________________________________________|
```

10.50 VCD—Vapor Compression Distillation Simulation Description

This routine simulates the performance of a vapor compression distillation (VCD) subsystem. It allows for calculation of product water quality, product water quantity, and power consumption. The principal user-specified performance parameters include constituent separation efficiencies, maximum allowable percent solids in the feed, average specific energy required, and water production rate versus percent solids characteristics. The routine allows for heat exchange between the component and its surroundings. The VCD does not generate flow; therefore, an active component must be upstream of the inlet. The VCD primary data base parameters are shown in table 82. The VCD constituent data base parameters are shown in table 83.



Discussion

The VCD water recovery characteristics are simulated using percent solid in feed versus water production rate empirical data available in the literature. The feed rate is assumed to be constant and is divided into the water production rate to obtain the fraction of the water in the feed that is distilled. The resulting relationship of percent solids versus the distilled fraction of water in the feed is linearized as shown in figure 75. The slope and Y-intercept of this line can be entered by the user through normal component parameter editing methods.

The carry-over of contaminants into the distillate stream is determined from user-defined separation efficiencies for each constituent. Separation efficiencies can be entered by the user through normal component parameter editing methods. The separation efficiency fraction for each constituent is given by the following equation:

$$E_s = 1 - \frac{\dot{m}_d X_d}{\dot{m}_i X_i} ,$$

where,

E_s = separation efficiency for individual constituents

$\dot{m}_d$ = VCD distillate stream total flow rate, lbm/h

$\dot{m}_i$ = VCD inlet waste total flow rate, lbm/h

X_d = constituent concentration in distillate stream, ppm

X_i = constituent concentration in inlet waste stream, ppm.

The maximum allowable percent solids in the feed can also be user-defined. In the case under consideration, if the percent solids in the feed reaches the maximum specified by the user, then all of the feed is diverted to the brine stream (no distillate is produced) and a message appears to the screen notifying the user of the condition.

Another user input is the specific energy requirements for the distillation process. The VCD subroutine multiplies the specific energy by the water production rate at a given time to yield the distillation power consumption. The accessories power is assumed to be constant and is definable by the user. Total power consumption is calculated by multiplying the specific energy by the water produced and adding to this value the constant accessories power. If the percent solids is above the maximum specified (resulting in a bypass condition) then the power consumption is set to the accessories power.

For simulation of thermal characteristics, the VCD is divided into a distillation section and a mean structure section. The mean structure section exchanges heat with its surroundings through convective and radiative pathways only (see fig. 77). To simplify the radiation heat transfer the radiation conductance is linearized internally. The user can input the proper convective and radiative heat transfer coefficients.

The distillation section is assumed to exchange heat with the mean structure section through conductive mechanisms only. The user can enter the appropriate conductive heat transfer coefficient. The distillation section is considered to transfer heat with the ambient through conductive mechanisms only. The user can input the desired coefficient for conductive heat transfer between the still and the cabin structure. It is assumed that the product water stream and the brine stream exit the VCD at the same temperature.

The VCD assumes a set outlet pressure for the brine and product water streams. This is considered to be a reasonable assumption since the VCD has integral fluid pumps that dictate the pressure of the outlet streams.

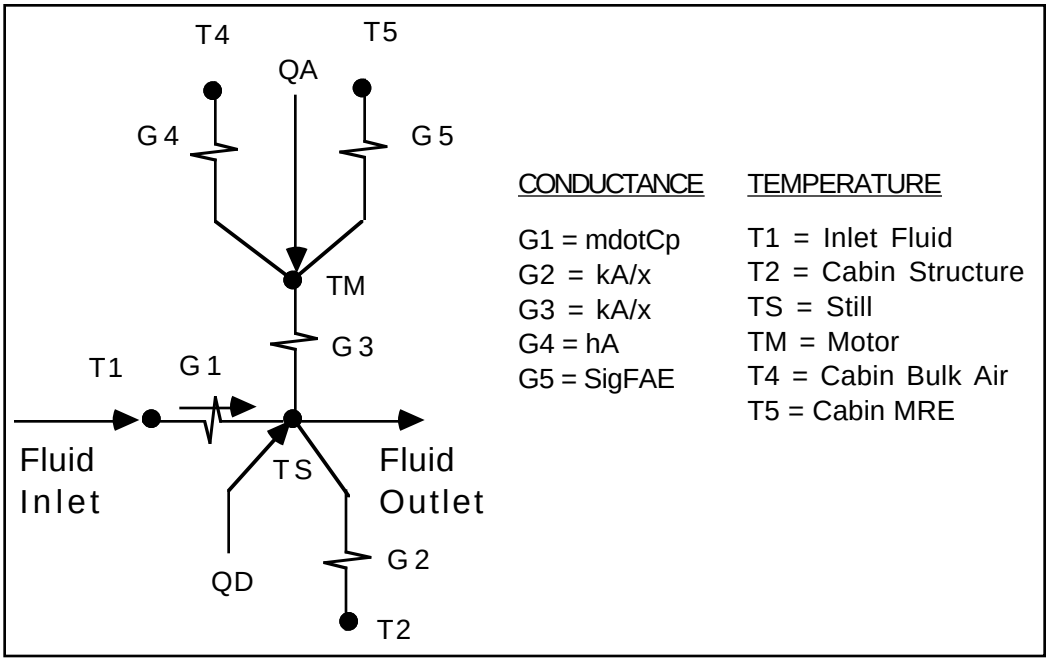


Figure 77. VCD heat transfer network.

Table 82. VCD edit screen.

```
CASE NAME: DEFAULT            VCD EDIT SCREEN              LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                     RECORD NUMBER: 000001
_____________________________________________________________________________
|            ** GENERAL INPUT DATA **          |  ** THERMAL INTERFACE DATA **  |
| 10 % SOLIDS SHUTDOWN POINT            50.00  |                                |
| 11 SLOPE %SOL VS FRAC H20 PROD LINE 0.00000  | 17 MOTOR H-A (BT/HR/F)   0.600 |
| 12 Y-INT %SOL VS FRAC H20 PROD LINE 0.20000  | 18 MOTR F-A-E   (FT^2)   2.500 |
|                                              | 19 MOTR-STIL G (B/H/F)    1.00 |
| 13 DISTLN SPEC ENERGY    (W-HR/LBM)   30.00  | 20 STIL-STRU G (B/H/F)    1.00 |
| 14 ACCESSORIES POWER CONSUMPTN  (W)   40.00  | 16 INIT MOTOR TEMP (F)   70.00 |
|                                              |     ** RESOURCE TRACKING **    |
| 21 PRODUCT STREAM PRESSURE   (PSIA)   20.00  | 7 POWER        (WATTS)    0.00 |
| 22 BRINE STREAM PRESSURE     (PSIA)   18.00  | 8 WEIGHT         (LBM)    0.00 |
|                                              | 9 VOLUME        (FT^3)    0.00 |
| CONSTITUENT EMPIRICAL DATA (^L TO OPEN)_     |                                |
|_____________________________________________|________________________________|
|                          ** BENCHMARK AND OUTPUT DATA **                     |
|                             MIN        NOM        MAX                        |
| WASTE IN FLOW  (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                      |
| BRINE OUT FLOW (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                      |
| PROD OUT FLOW  (LBM/HR)  0.00E+00   0.00E+00   0.00E+00                      |
| POWER CONSUMED  (WATTS)  0.00E+00   0.00E+00   0.00E+00                      |
| 75 FINAL MOTOR TEMP (F)    0.00                                              |
|_____________________________________________________________________________|
```

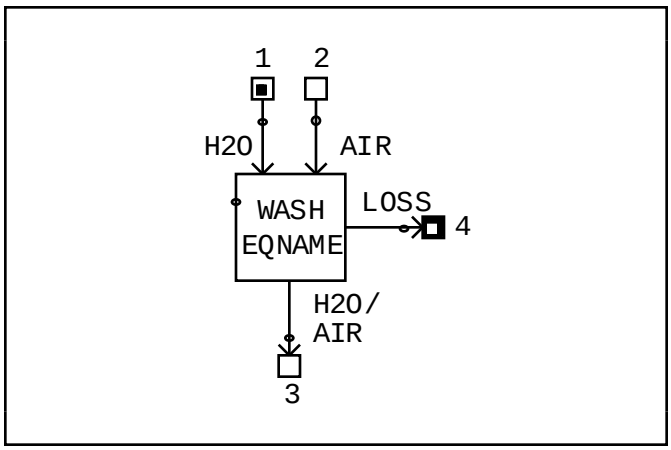
Table 83. VCD constituent edit screen.

CASE NAME: DEFAULT VCD CONSTITUENT SCREEN LAST UPDATED: 890215
SUBSYSTEM: COMPONENT: RECORD NUMBER: 000001

LABEL	SEP EFFCY FRACTION	LABEL	SEP EFFCY FRACTION	LABEL	SEP EFFCY FRACTION
CONSTIT 2	1.0000000	CONSTIT 19	1.0000000	CONSTIT 36	1.0000000
CONSTIT 3	1.0000000	CONSTIT 20	1.0000000	CONSTIT 37	1.0000000
CONSTIT 4	1.0000000	CONSTIT 21	1.0000000	CONSTIT 38	1.0000000
CONSTIT 5	1.0000000	CONSTIT 22	1.0000000	CONSTIT 39	1.0000000
CONSTIT 6	1.0000000	CONSTIT 23	1.0000000	CONSTIT 40	1.0000000
CONSTIT 7	1.0000000	CONSTIT 24	1.0000000	CONSTIT 41	1.0000000
CONSTIT 8	0.0000000	CONSTIT 25	1.0000000	CONSTIT 42	1.0000000
CONSTIT 9	1.0000000	CONSTIT 26	1.0000000	CONSTIT 43	1.0000000
CONSTIT 10	1.0000000	CONSTIT 27	1.0000000	CONSTIT 44	1.0000000
CONSTIT 11	1.0000000	CONSTIT 28	1.0000000	CONSTIT 45	1.0000000
CONSTIT 12	1.0000000	CONSTIT 29	1.0000000	CONSTIT 46	1.0000000
CONSTIT 13	1.0000000	CONSTIT 30	1.0000000	CONSTIT 47	1.0000000
CONSTIT 14	1.0000000	CONSTIT 31	1.0000000	CONSTIT 48	1.0000000
CONSTIT 15	1.0000000	CONSTIT 32	1.0000000	CONSTIT 49	1.0000000
CONSTIT 16	1.0000000	CONSTIT 33	1.0000000	CONSTIT 50	1.0000000
CONSTIT 17	1.0000000	CONSTIT 34	1.0000000		
CONSTIT 18	1.0000000	CONSTIT 35	1.0000000		

10.51 WASH—WASH Process Simulation Description

The WASH routine simulates a generalized wash process. The routine models the consumption of water and the associated introduction of contaminants into the outlet streams for a clothes, hand, shower, or dish washing apparatus. An active source of air must also be connected to the air inlet stream of the WASH process. It should be understood that addition of the previously mentioned contaminants to the WASH outlet streams creates a minor system imbalance (outlet flows greater than inlet flows). The user can specify the quantity and nature of these contaminants in the input data definition file. The user sets the water consumption demand for the WASH. The number of washes per day is also specified by the user.



10.51.1 General Discussion

The two outlet streams associated with the wash process are the normal drain (stream 3) and the loss-to-cabin stream (stream 4). Stream 3 is the mixture of the inlet air, water, and introduced contaminants minus the loss due to leakage (stream 4) determined by the leakage loss constant. This constant is merely a percentage of the inlet total flow that is lost.

The temperatures of the outlet streams are assumed to be equal and are calculated from an energy balance equation on the two inlet streams combined:

$$T_o = \frac{\sum_{j=1}^{n} \dot{m}_j C_{p_j} T_j}{\sum_{j=1}^{n} \dot{m}_j C_{p_j}},$$

where,

T_o = outlet stream temperature (°F)

T_j = inlet stream j temperature (°F)

$\dot{m}_j$ = inlet stream j mass flow (lbm/h)

C_{p_j} = inlet stream j specific heat (Btu/lbm/°F)

n = number of inlet streams.

The external heat transfer is determined via ten input parameters that are utilized by the generic thermal network solution described in section 8.3.3.

10.51.2 Pressure Calculations

The outlet stream pressures are considered to be equal and are based on a flow resistance constant specified by the user. See section 8.2.5 for a detailed discussion of the pressure drop calculations.

10.51.3 Input Parameters

The user input parameters for the WASH routine are shown in table 84. The constituents data base parameters are shown in table 85.

Table 84. WASH edit screen.

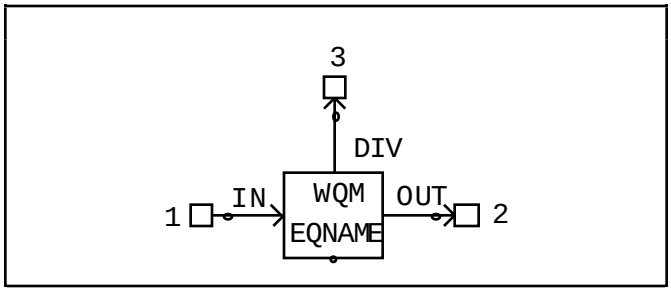
```
CASE NAME: DEFAULT                                   WASH EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                  RECORD NUMBER: 000001
_____________________________________________________________________________________________________________________________
|            ** GENERAL INPUT DATA **          |    ** THERMAL CHARACTERISTICS DATA **     |     ** MISCELLANEOUS  PARAMETERS **      |
| 10 NUMBER OF WASHES PER DAY          6.000   | 14 SHELL-ENV CONVCT (BTU/HR/F)      0.000 | 24 ** OPEN INPUT LOCATION **      0.000 |
| 11 WATER DEMAND     (LBM/WASH)       8.000   | 15 SHELL-ENV RAD  FAE    (FT^2)     0.000 | 25 ** OPEN INPUT LOCATION **      0.000 |
| 12 LEAKAGE LOSS   (% OF INLET)       2.000   | 16 SHELL-ENV CONDCT (BTU/HR/F)      0.000 | 75 ** OPEN INPUT LOCATION **      0.000 |
| 13 FLOW RES FACTR (K=DP/XM**2)  0.200        | 17 MASS-SHEL CONDCT (BTU/HR/F)      1.000 |                                         |
|                                              | 18 WALL-MASS CONDCT (BTU/HR/F)      1.000 |          ** RESOURCE TRACKING **        |
|                                              | 19 FLUID-WALL EFFECT   (0<X<1) 0.9500000  | 7  POWER              (WATTS)     0.000 |
|                                              | 20 THERMAL CAPACITNCE  (BTU/F)      0.000 | 8  WEIGHT               (LBM)     0.000 |
|                                              | 21 INITIAL MASS TEMPRTURE  (F)     75.000 | 9  VOLUME              (FT^3)     0.000 |
|                                              | 22 THERMAL RELAXATN CRITERIA    0.0000100 |                                         |
| -- CONSTITUENT EMP DATA  (^L  TO OPEN ) _    | 23 THERMAL MAX ITERATN COUNT      100.000 |                                         |
|_____________________________________________|_____________________________________|_________________________________________|
|                        ** BENCHMARK DATA **                  |                    ** OUTPUT DATA **                                |
|                          MIN          NOM          MAX        | 76  PRESENT MASS TEMPERATURE                (F)     0.000          |
|   WASTE OUT FLOW (LBM/HR)     0.000        0.000        0.000 | 77  PRESENT SHELL TEMPERATURE               (F)     0.000          |
|   LEAKAGE FLOW   (LBM/HR)     0.000        0.000        0.000 | 78  **  OPEN OUTPUT LOCATION  **                    0.000          |
|   WATER IN FLOW  (LBM/HR)     0.000        0.000        0.000 |                                                                     |
|   CIRC AIR FLOW  (LBM/HR)     0.000        0.000        0.000 |                                                                     |
|_____________________________________________________________|_____________________________________________________________________|
```

Table 85. WASH constituents edit screen.

```
CASE NAME: DEFAULT        WASH CONSTITUENT SCREEN          LAST UPDATED: 890215
SUBSYSTEM:                   COMPONENT:                   RECORD NUMBER: 000001
_____________________________________________________________________________________
|    LABEL    |    INPUT    |    LABEL    |    INPUT    |    LABEL    |    INPUT    |
|             |   LBM/WASH  |             |   LBM/WASH  |             |   LBM/WASH  |
|_____________|_____________|_____________|_____________|_____________|_____________|
| OXYGEN      |   0.000000  | CONSTIT 19  |   0.000000  | CONSTIT 36  |   0.000000  |
| NITROGEN    |   0.000000  | CONSTIT 20  |   0.000000  | CONSTIT 37  |   0.000000  |
| CARBIN DIOX |   0.000000  | CONSTIT 21  |   0.000000  | CONSTIT 38  |   0.000000  |
| HYDROGEN    |   0.000000  | CONSTIT 22  |   0.000000  | CONSTIT 39  |   0.000000  |
| CARBON (S)  |   0.000000  | CONSTIT 23  |   0.000000  | CONSTIT 40  |   0.000000  |
| METHANE     |   0.000000  | CONSTIT 24  |   0.000000  | CONSTIT 41  |   0.000000  |
| WATER       |   0.000000  | CONSTIT 25  |   0.000000  | CONSTIT 42  |   0.000000  |
| FREON-11    |   0.000000  | CONSTIT 26  |   0.000000  | CONSTIT 43  |   0.000000  |
| CONSTIT 10  |   0.000000  | CONSTIT 27  |   0.000000  | CONSTIT 44  |   0.000000  |
| CONSTIT 11  |   0.000000  | CONSTIT 28  |   0.000000  | CONSTIT 45  |   0.000000  |
| CONSTIT 12  |   0.000000  | CONSTIT 29  |   0.000000  | CONSTIT 46  |   0.000000  |
| CONSTIT 13  |   0.000000  | CONSTIT 30  |   0.000000  | CONSTIT 47  |   0.000000  |
| CONSTIT 14  |   0.000000  | CONSTIT 31  |   0.000000  | CONSTIT 48  |   0.000000  |
| CONSTIT 15  |   0.000000  | CONSTIT 32  |   0.000000  | CONSTIT 49  |   0.000000  |
| CONSTIT 16  |   0.000000  | CONSTIT 33  |   0.000000  | CONSTIT 50  |   0.000000  |
| CONSTIT 17  |   0.000000  | CONSTIT 34  |   0.000000  |             |             |
| CONSTIT 18  |   0.000000  | CONSTIT 35  |   0.000000  |             |             |
|_____________|_____________|_____________|_____________|_____________|_____________|
```

10.52 WQM—Water Quality Monitor Simulation Description

The WQM routine simulates the performance of an idealized water quality monitoring and control system that consists of a monitoring unit that controls an integral diverter valve based on the quality of the entering feed stream. It is assumed that the unit is able to monitor all constituents in the feed stream on a continuous basis and that no reagents are required.



10.52.1 General Discussion

The routine compares the inlet concentration of each constituent in the feed stream to the user-defined reference concentration for that constituent at each time step during the simulation. If any one inlet concentration is greater than the input value, the flow is routed to the divert stream based on a user-defined time lag constant. This is done by splitting the flow between the divert stream (stream 3) and the pass-through stream (stream 2) based on the ratio of the time lag constant to the simulation time step by the following equations:

$$\dot{m}_2 = \dot{m}_1(K/\Delta t)\,,$$

$$\dot{m}_3 = \dot{m}_1 - \dot{m}_2\,,$$

where,

$\dot{m}_1$ = inlet mass flow rate (lbm/h)

$\dot{m}_2$ = pass-through mass flow rate (lbm/h)

$\dot{m}_3$ = diverted mass flow rate (lbm/h)

K = time lag constant (h)

Δt = simulation time step (h).

If the quantity $K/\Delta t$ is greater than or equal to 1.0, all flow will be passed through to stream 2 regardless of the inlet composition. This case occurs when the time lag constant is greater than or equal to the simulation time step.

The heat transfer with the surroundings is determined from user provided mean convective, radiative, and structural conduction coefficients for external heat transfer along with a coefficient that describes the heat transfer between the flowing fluid and the internal flow path surfaces.

The pressure drop through the unit is determined from the user-defined hydraulic equivalent length and diameter. See section 8.2.5 for an overview concerning the determination of these values.

10.52.2 Input Parameters

The user input parameters for the WQM routine are shown in table 86. The constituents data base parameters are shown in table 87.

Table 86. WQM edit screen.

```
CASE NAME: DEFAULT                                    WQM EDIT SCREEN                                  LAST UPDATED: 890215
SUBSYSTEM:                COMPONENT:                                                                   RECORD NUMBER: 000001
|____________________________________________________________________________________________________________________________|
|          ** GENERAL INPUT DATA **          |     ** THERMAL CHARACTERISTICS DATA **     |     ** MISCELLANEOUS  PARAMETERS **      |
| 10 TIME LAG CONSTANT       (HR)     0.000 | 13 SHELL-ENV CONVCT (BTU/HR/F)     0.000 | 23 ** OPEN INPUT LOCATION **      0.000 |
| 11 HYDRAUL EQUIV LENGTH    (FT)    10.000 | 14 SHELL-ENV RAD  F*A*E (FT^2)     0.000 | 24 ** OPEN INPUT LOCATION **      0.000 |
| 12 HYDRAUL EQUIV DIAMETER  (IN)     1.000 | 15 SHELL-ENV CONDCT (BTU/HR/F)     0.000 | 74 ** OPEN INPUT LOCATION **      0.000 |
|                                           | 16 MASS-SHEL CONDCT (BTU/HR/F)     1.000 |                                           |
|                                           | 17 WALL-MASS CONDCT (BTU/HR/F)     1.000 |          ** RESOURCE TRACKING **          |
|                                           | 18 FLUID-WALL EFFECT   (0<X<1) 0.9500000 | 7  POWER              (WATTS)     0.000 |
|                                           | 19 THERMAL CAPACITNCE  (BTU/F)     0.000 | 8  WEIGHT               (LBM)     0.000 |
|                                           | 20 INITIAL MASS TEMPRTURE  (F)    75.000 | 9  VOLUME              (FT^3)     0.000 |
|                                           | 21 THERMAL RELAXATN CRITERIA   0.0000100 |                                           |
| CONSTIT DIVERT CONCENTRTNS (^L TO OPEN) _ | 22 THERMAL MAX ITERATN COUNT     100.000 |                                           |
|____________________________________________|____________________________________|____________________________________________|
|                      ** BENCHMARK DATA **                                  |                  ** OUTPUT DATA **                        |
|                                                                            |  75  PRESENT PRESSURE DROP              (PSID)     0.000  |
|   NOTE : Maximum constituent inlet concentrations that ocurred             |  76  PRESENT MASS FLOW RATE           (LBM/HR)     0.000  |
|          for the last completed simulation may be accessed by              |  77  PRESENT MASS  TEMPERATURE             (F)     0.000  |
|          selecting the constituents' subscreen toggle listed               |  78  PRESENT SHELL TEMPERATURE             (F)     0.000  |
|          in the upper left-hand segment of this editing tem-               |  79  **  OPEN OUTPUT LOCATION  **                  0.000  |
|          plate.                                                            |                                                           |
|____________________________________________________________________________|____________________________________________________________|
```

Table 87. WQM constituents edit screen.

CASE NAME: DEFAULT — WQM CONSTITUENT SCREEN — LAST UPDATED: 890215
SUBSYSTEM: DEFAULT COMPONENT: — RECORD NUMBER: 000001

LABEL	DIVERT CONC PPM	MAX IN CONC PPM	LABEL	DIVERT CONC PPM	MAX IN CONC PPM	LABEL	DIVERT CONC PPM	MAX IN CONC PPM
OXYGEN	2000000.0	0.0	CONSTIT 19	2000000.0	0.0	CONSTIT 36	2000000.0	0.0
NITROGEN	2000000.0	0.0	CONSTIT 20	2000000.0	0.0	CONSTIT 37	2000000.0	0.0
CARBON DIOX	2000000.0	0.0	CONSTIT 21	2000000.0	0.0	CONSTIT 38	2000000.0	0.0
HYDROGEN	2000000.0	0.0	CONSTIT 22	2000000.0	0.0	CONSTIT 39	2000000.0	0.0
CARBON (S)	2000000.0	0.0	CONSTIT 23	2000000.0	0.0	CONSTIT 40	2000000.0	0.0
METHANE	2000000.0	0.0	CONSTIT 24	2000000.0	0.0	CONSTIT 41	2000000.0	0.0
WATER	2000000.0	0.0	CONSTIT 25	2000000.0	0.0	CONSTIT 42	2000000.0	0.0
FREON-11	2000000.0	0.0	CONSTIT 26	2000000.0	0.0	CONSTIT 43	2000000.0	0.0
CONSTIT 10	2000000.0	0.0	CONSTIT 27	2000000.0	0.0	CONSTIT 44	2000000.0	0.0
CONSTIT 11	2000000.0	0.0	CONSTIT 28	2000000.0	0.0	CONSTIT 45	2000000.0	0.0
CONSTIT 12	2000000.0	0.0	CONSTIT 29	2000000.0	0.0	CONSTIT 46	2000000.0	0.0
CONSTIT 13	2000000.0	0.0	CONSTIT 30	2000000.0	0.0	CONSTIT 47	2000000.0	0.0
CONSTIT 14	2000000.0	0.0	CONSTIT 31	2000000.0	0.0	CONSTIT 48	2000000.0	0.0
CONSTIT 15	2000000.0	0.0	CONSTIT 32	2000000.0	0.0	CONSTIT 49	2000000.0	0.0
CONSTIT 16	2000000.0	0.0	CONSTIT 33	2000000.0	0.0	CONSTIT 50	2000000.0	0.0
CONSTIT 17	2000000.0	0.0	CONSTIT 34	2000000.0	0.0			
CONSTIT 18	2000000.0	0.0	CONSTIT 35	2000000.0	0.0			

APPENDIX A

SAMPLE PROBLEMS

A.1 INTRODUCTION

The sample problems presented in this section should assist the new user in obtaining a working knowledge of the CASE/A modeling package. The user is encouraged to review the problems carefully while actually working at a computer terminal. Whenever possible, references are made to the appropriate sections of the manual so that additional information may be readily available. The first problem provides a step-by-step illustration of the commands necessary to construct and solve a simple active thermal control (ATC) problem. A plotset is created using the Integrated Plot Utility (IPU) and the steps necessary for obtaining graphics hardcopy plots are presented. The second problem focuses on the utilization of user-defined operations logic (OPS logic) and the constants data base (USERCON) through the development of a water conditioning model. The OPS logic is provided in the text and the CASE/A functions used in the code are described as they appear. The problems are intended to complement one another and duplicate information is kept to a minimum. The new user will find that both a careful review of the problems plus a thorough study of all referenced text are essential to quickly become proficient with the CASE/A modeling package.

A.2 SAMPLE PROBLEM NO. 1

The following sample problem introduces the new user to CASE/A through the construction of a simplified model designed to simulate an ATC loop. A description of all commands used for subsystem screen construction and data base management as well as solution control and execution is provided. The user must also be operating on a graphics terminal (Tektronix 4014 or compatible) during the construction of the case. Consult appendix C, CASE/A Installation Notes, for additional information on CASE/A linking and execution procedures.

A.2.1 Subsystem Assembly

The ATC model constructed in this example consists of the following CASE/A components: pump, coldplate, and thermal bus. Additionally, a controller component will be used to monitor a specified temperature and adjust the flow rate accordingly. Complete descriptions of all the components are provided in section 10, Component Library Descriptions. To begin the assembly of a model, the user must first enter the CASE/A environment by executing the following command at the DCL prompt:

```
$ RUN/NODEB CASEA$CODE:CASEA
```

The user first specifies that a new case is to be constructed and provides a case name. This is accomplished at the CASE/A command prompt:

```
CASEA > NEWCASE;ATC
```

The user has the option to use either a semicolon ";" or blank space as the delimiter in a CASE/A command line. The semicolon delimiter is used throughout this manual for clarity. Also, all commands are listed alphabetically in appendix B if further reference is desired. CASE/A indicates that an active case has been loaded (even though the case may be empty, as it is now) by changing the prompt to include the case name, as shown below. The user must now indicate that a subsystem is to be created by issuing the `SS` command. CASE/A currently supports up to 100 subsystems in a single case. This allows large cases to be assembled in a modular fashion thereby simplifying construction by reducing developmental phases to smaller, more manageable portions. If multiple subsystems are to be created, the user should assign

logical, unique names to each one as it is created. If only one subsystem is to be created, it is frequently given the same name as the case:

```
CASEA_ATC > SS;ATC
```

An empty subsystem screen will now appear with the case name, subsystem name, and cabin assignment at the top of the screen. Since this subsystem has not been assigned to a cabin, **NONE** should appear next to cabin assignment. CABIN assignments are performed to provide a thermal environment (other than the default environment of 75 °F for structural, convective, and radiant boundary temperatures) and are explained in section 6.5. It is through this graphical interface that the components are positioned on the screen and connected to represent the actual hardware configuration. Using figure A-1 as a guide, the subsystem is assembled as follows:

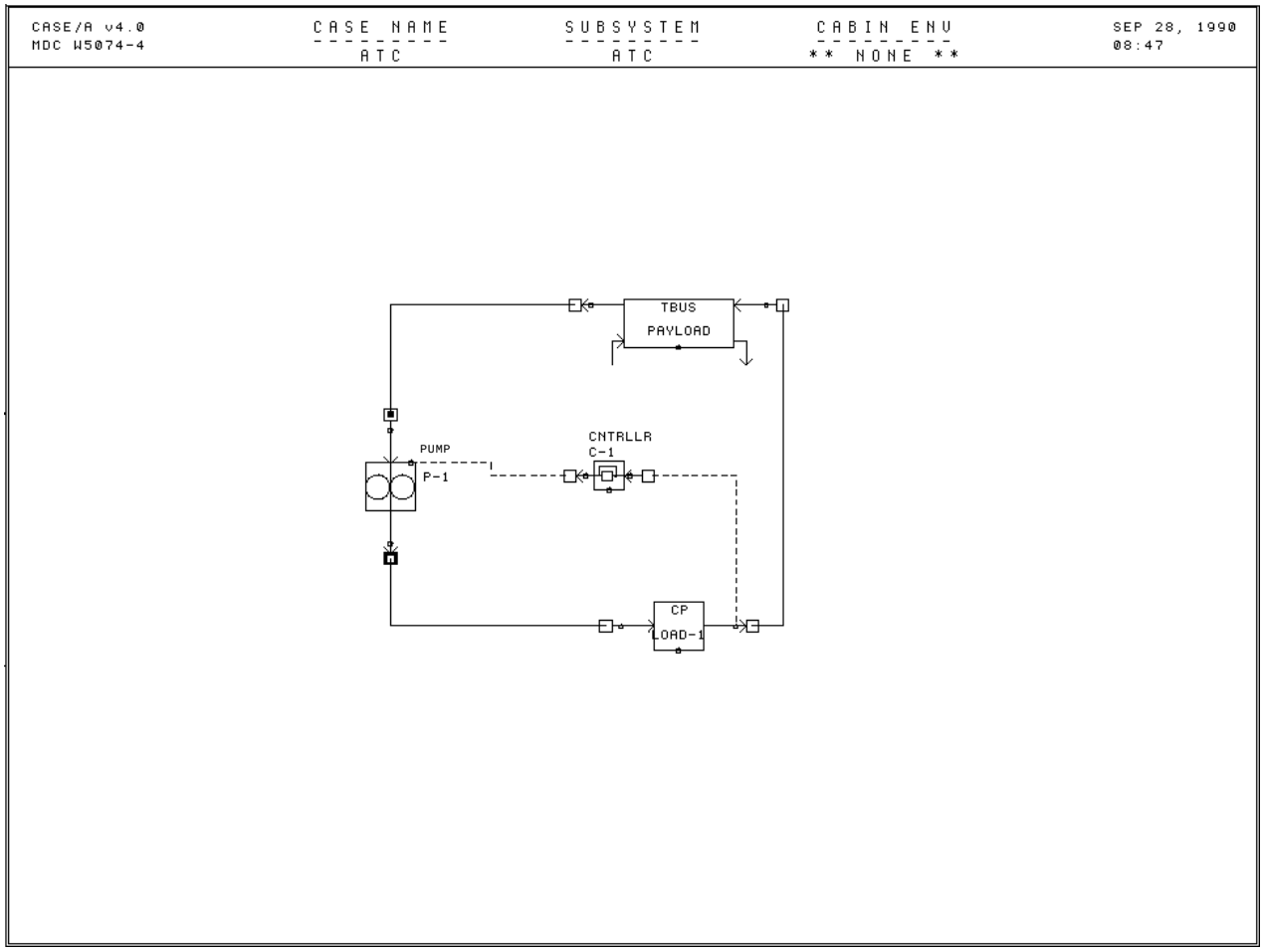


Figure A-1. Sample problem 1: subsystem schematic.

A. Locate a PUMP component using the LOcate command as indicated:

```
CASEA_ATC > LO;PUMP;P-1
```

The first parameter in the argument list of the `LO` command represents the type of component to be located. The last argument is the user specified component name, and is limited to eight (8) characters. No two components, regardless of type, may have the same name in any given case. An exception to this rule is the BUBBLE component. Due to its function, duplicate names are required (see section 10.7). Upon execution of the `LO` command, the cross-hairs will appear on the screen. Position the pump on the screen by moving the cross-hairs to the desired location and hitting any key. The pump

component icon will then be drawn at that location in the default orientation, i.e., horizontally with flow directed left to right. Continue in this manner to locate the remaining components.

B. Locate a CP (cold plate) component and name it LOAD-1:

```
CASEA_ATC > LO;CP;LOAD-1
```

C. Locate a TBUS (thermal bus) component and name it PAYLOAD:

```
CASEA_ATC > LO;TBUS;PAYLOAD
```

D. Locate a CNTRLLR (controller) component and name it C-1:

```
CASEA_ATC > LO;CNTRLLR;C-1
```

At this point, it is apparent that the subsystem schematic could be more logically presented if the PUMP, CNTRLLR, and the TBUS components were rotated using the `ROTATE` command:

```
CASEA_ATC > ROTATE;-90
```

The argument for the `ROTATE` command must be in multiples of 90˚ and with a magnitude not greater than 360˚. A negative value represents a clockwise rotation. Upon execution of the `ROTATE` command, the crosshairs will appear. Place the crosshairs on the component to be rotated (the pump) and hit any key. The component will be shown in its new orientation upon execution of the next `RD` (redraw) command:

```
CASEA_ATC > RD
```

Similarly, rotate the CNTRLLR component 180˚:

```
CASEA_ATC > ROTATE;180
```

Place the crosshairs on the CNTRLLR component and hit any key. Continuing, rotate the TBUS component 180˚. Execute `RD` to show the final orientation of the components.

E. The component streams are connected using the `CN` (connect) command as follows:

```
CASEA_ATC > CN[;LEQ;DEQ]
```

The LEQ and DEQ arguments are optional and represent equivalent length in feet and equivalent diameter in hundredths of an inch for hydraulic flow calculations. If these parameters are omitted, values of 1 foot and 6 inches are selected for LEQ and DEQ, respectively. Upon execution of the CN command, the crosshairs will appear. The starting point of a connection path is identified by placing the crosshairs in the “hit box” of the desired component stream and typing the character “S” (start). The crosshairs will remain active until the ending point of the connection path is identified. This is performed by moving the crosshairs to the “hit box” of the desired component stream and typing the character “E” (end). If, due to component positioning, it becomes necessary to segment the connection stream as in the example figure, type the character “I” (intermediate) at the desired point. This leaves the crosshairs active and the connection path may be redirected as needed.

The user should have a clear understanding of the difference between a “connection path” and a “component stream”. A connection path joins two components and has associated with it a hydraulic length and diameter for pressure drop calculations. A component stream is simply an inlet/outlet to that component and is part of the component icon as clearly shown when a component is first `LO`cated on a subsystem screen. The CNTRLLR component is connected with the `CN` command but the “hit boxes”

of the controller are connected to “hit dots” of the desired component(s). Note that there are “hit dots” on the body of a component as well as on all streams associated with that component. A complete description of both types (body and stream) of “hit dots” is presented in section 2.3. The connection between a controller “hit box” and a body/stream “hit dot” is termed a “data path”. A data path appears on the screen as a dotted line. It is left to the user to connect the components as shown in the figure.

A.2.2 Data Base Management

After a component has been located on a subsystem screen, its performance parameters may be modified to match the user’s specific requirements. The manner in which a component data base is edited depends on the type of terminal being used. As mentioned previously, subsystem assembly may be performed only on a graphics terminal. Data base management, however, may be performed on a graphics terminal or a text terminal, with the latter being the preferred choice. The user should refer to section 6.1 for assistance in specifying the terminal type and section 3.0 for a complete discussion of the editing functions. The following instructions are applicable to users working on a text terminal, specifically a DEC VT100 or compatible, which takes advantage of the full screen editing mode. Users working on a graphics-only terminal, namely Tektronix or compatible, should read the following to learn the basic concepts and then refer to the end of this section for help with specific commands. First, edit the pump data base:

```
CASEA_ATC > ED;PUMP
```

If more than one pump existed in the case, the user would be requested to select one. Otherwise, the pump edit screen will appear, as shown in figure A-2. The arrow keys allow movement within the edit screen. Up/down arrows move the cursor between data fields and left/right arrows within a single data field. All data entry occurs in the overstrike mode, so care should be taken with respect to decimal placement and residual digits from overwritten data. Using the arrow keys, position the cursor in the data field corresponding to the mass flow rate (item 10) and set the rate to 50 lb/h. Next, change the outlet pressure to 20.0 lb/in^2 absolute, and the fluid code to 8 (8 = water, see section 7.2). To exit from any edit screen and save the changes, simply enter CNTRL-Z. To exit without saving changes, enter CNTRL-X (Escape).

Similarly, edit the thermal bus (`ED;TBUS`) and set the control fluid temperature to 35 ˚F. Next, set the thermal load in the cold plate (`ED;CP`) to 1000 watts. The default edit screens for these components are shown in figures A-3 and A-4. Each component has a previously defined set of default performance parameters. Any of these values may be accepted or modified, as desired. The default values usually represent *Spacelab* or space station data and will most likely require modification to match specific performance data. The values chosen herein are purely arbitrary and are only for purposes of illustration. Note that the responsibility of selecting realistic values for any of the component parameters relies solely on the user.

The CNTRLLR data base is empty by default, i.e. no control functions are defined. In this example, the controller component is used to adjust the flow rate of the pump based on fluid temperature at the outlet of the CP. Specifically, proportional control logic is used to maintain the fluid temperature at the outlet of the cold plate at a 70 ˚F setpoint temperature (±3 ˚F). The controller component data base is modified in the same manner as the other components (`ED;CNTRLLR`) with one basic exception. Instead of changing performance parameters, as above, the user specifies an operation(s) list to be applied to the input data path or output data path of the controller. Modify the CNTRLLR to match the screen shown in figure A-5. The controller commands used are fully explained in section 10.13.

```
CASE NAME: ATC                                   PUMP EDIT SCREEN                                   LAST UPDATED: 891005
SUBSYSTEM: ATC          COMPONENT: P-1                                                            RECORD NUMBER: 000335
_____________________________________________________________________________________________________________________
|              * MISC INPUT DATA *              |    * THERMAL CHARACTERISTICS DATA *     |   * CENTRIFUGAL PUMP OPTION CONSTANTS *  |
| 10 MASS FLOWRATE          (LBM/HR)    50.000 | 22 SHEL-ENV CONVCT (BTU/H/F)     0.200 | 35 FLOW DAMPING FACT (0<X<1)   0.10000000 |
| 11 INLET PRESSURE          (PSIA)      0.000 | 23 SHEL-ENV RADIAT FAE (FT2)     0.800 | 13 CHARCTRISTIC CURVE CON A0  0.00000E+00 |
| 12 OUTLET PRESSURE         (PSIA)     20.000 | 24 SHEL-ENV CONDCT (BTU/H/F)     1.000 | 14 CHARCTRISTIC CURVE CON A1  0.00000E+00 |
| 18 IMPELLER EFFICIENCY    (0<X<1)      0.700 | 25 MASS-SHEL COND  (BTU/H/F)     1.000 | 15 CHARCTRISTIC CURVE CON A2  0.00000E+00 |
| 19 MOTOR EFFICIENCY       (0<X<1)      0.700 | 26 WALL-MASS COND  (BTU/H/F)    50.000 | 16 CHARCTRISTIC CURVE CON A3  0.00000E+00 |
| 20 SPECIFIED MOTR QFLAG (0=N,1=Y)      0.000 | 27 FLUID-WALL EFFECT (0<X<1)     0.900 | 17 CHARCTRISTIC CURVE CON A4  0.00000E+00 |
| 21 SPECIFIED MOTOR QLOAD (BTU/HR)      0.000 | 28 THERMAL CAPACTNCE (BTU/F)     0.000 | 36 ** OPEN INPUT LOCATION **          0.000 |
| 34 FLUID CODE(0=?,2-50=CONSTIT #)      8.000 | 29 INITIAL MASS TEMP     (F)    75.000 | 7  POWER                 (WATTS)      0.001 |
| 32 ** OPEN INPUT LOCATION **           0.000 | 30 THERMAL RELAX CRITERIA    0.0000100 | 8  WEIGHT                 (LBM)       0.000 |
| 33 ** OPEN INPUT LOCATION **           0.000 | 31 THERMAL SOLUTN MAX COUNT    100.000 | 9  VOLUME                (FT^3)       0.000 |
|_____________________________________________|_________________________________________|_____________________________________|
|                        * BENCHMARK DATA *                                            |                  * OUTPUT DATA *                  |
|                                  MIN         NOM         MAX                         | 37  MAX CAPACITY FLOW RATE       (LBM/HR)    50.000 |
|  PUMP INLET PRESS  (PSIA)    19.9878     19.9902     19.9908                         | 38  PRESENT PRESSURE RISE         (PSID)      0.000 |
|  PUMP OUTLET PRESS (PSIA)    20.0000     20.0000     20.0000                         | 39  PRESENT FLOW CONVERGENCE FLAG                 0.000 |
|  PUMP PRESS RISE   (PSIA)     0.0092      0.0098      0.0122                         | 40  PRESENT MASS TEMPERATURE          (F)    43.406 |
|  PUMP FLOW RATE  (LBM/HR)    50.0000     50.0001     50.0000                         | 41  PRESENT SHELL TEMPERATURE         (F)    64.518 |
|  PUMP POWER       (WATTS)     0.0008      0.0009      0.0011                         | 42  ** OPEN OUTPUT LOCATION **                  0.000 |
|_____________________________________________________________________________________|_____________________________________|
```

Figure A-2. Sample problem 1: PUMP edit screen.

```
CASE NAME: ATC                 TBUS EDIT SCREEN            LAST UPDATED: 890929
SUBSYSTEM: ATC               COMPONENT: PAYLOAD           RECORD NUMBER: 000075
_____________________________________________________________________________
| ** INPUT DATA                                                             |
|                                                   * RESOURCE TRACKING *   |
|10 EFFECTIVITY                              0.90000 |7 POWER    (WATTS)    0.00 |
|11 CONTROL FLUID TEMPERATURE        (F)    35.00 |8 WEIGHT     (LBM)    0.00 |
|                                                   |9 VOLUME    (FT^3)   0.000 |
|          * PRESSURE DROP PARAMETERS *             |                           |
|12 EQUIVALENT LENGTH               (FT)    10.00 |                           |
|13 EQUIVALENT DIAMETER             (IN)    0.500 |                           |
|___________________________________________________|___________________________|
| ** BENCHMARK DATA                                 | ** OUTPUT DATA            |
|                        MIN         NOM         MAX                            |
| OUTLET TEMP  (F)    37.         41.         43.  |14 HEAT TRNSFR (W)   990.98 |
| HEAT TRANSFR (W)  0.24E+03   0.81E+03   0.99E+03|                           |
|___________________________________________________|___________________________|
```

Figure A-3. Sample problem 1: TBUS edit screen.

```
CASE NAME: ATC                  CP EDIT SCREEN             LAST UPDATED: 890921
SUBSYSTEM: ATC                COMPONENT: LOAD-1            RECORD NUMBER: 000054
_____________________________________________________________________________
| ** INPUT DATA                                                             |
| 10 THERMAL LOAD         (WATTS)  1000.00 |     * RESOURCE TRACKING *       |
| 11 EFFECTIVENESS                    0.90 | 7 POWER           (WATTS)    0.00 |
|                                          | 8 WEIGHT            (LBM)    0.00 |
| 13 INITIAL MASS TEMP        (F)    70.00 | 9 VOLUME           (FT^3)    0.00 |
| 16 THERMAL MASS         (BTU/F)    10.40 |                                 |
|                                          |    * THERMAL INTERFACE DATA *   |
|       * PRESSURE DROP PARAMETERS *       | 19 SHEL-ENV HA (BTU/HR/F)    0.00 |
| 14 EQUIVALENT LENGTH       (FT)   10.000 | 20 SHEL-ENV FAE    (FT^2)    0.00 |
| 15 EQUIVALENT DIAMETER     (IN)    0.500 | 21 SHL-ENV KA/X  (B/HR/F)    0.00 |
|                                          | 22 MAS-SHL KA/X  (B/HR/F)    1.00 |
|        * THERMAL CONDUCTANCE DATA *      |                                 |
| 17 CONTACT AREA          (FT^2)   2.0000 | 23 THERM RELAX CRITERIA .1000E-04 |
| 18 CONTACT COND (BTU/HR/FT^2/F)   20.000 | 24 THERM MAX LOOP           100.0 |
|__________________________________________|_________________________________|
| ** BENCHMARK DATA                                 | ** OUTPUT DATA            |
|                        MIN         NOM         MAX                            |
|  BASEPLT TEMP (F)   55.      0.10E+03   0.12E+03 | 27 BASEPL TMP (F) .12E+03 |
|  EQUIPMT TEMP (F)   74.      0.17E+03   0.20E+03 | 28 EQUIP TEMP (F) .20E+03 |
|___________________________________________________|___________________________|
```

Figure A-4. Sample problem 1: CP edit screen.

```
CASE NAME: ATC                                   CNTRLLR EDIT SCREEN                                LAST UPDATED: 891004
SUBSYSTEM: ATC          COMPONENT: C-1                                                            RECORD NUMBER: 000093
_____________________________________________________________________________________________________________________
| ** INPUT DATA                                 INPUT ITEM CODE (ARRAY LOCATION)      2.0 |    POWER      (WATTS)   0.000 |
|       ACTIVATION CODE (0 < C < 7)       4.0   OUTPUT ARRAY CODE (1-C,2-P,3-K)       3.0 |    WEIGHT       (LBM)   0.000 |
|       INPUT ARRAY CODE  (1-C,2-P,3-K)   2.0   OUTPUT ITEM CODE (ARRAY LOCATION)     10.0 |    VOLUME      (FT^3)   0.000 |
|_____________________________________________________________________________________________|_____________________________|
| *PROGRAMMING STEPS *  9                 | 18 XXXXXXXXXXXXXXX | 27 XXXXXXXXXXXXXXX | 36 XXXXXXXXXXXXXXX | 44 XXXXXXXXXXXXXXX |
| 1 PC:70:5:3           | 10              | 19 XXXXXXXXXXXXXXX | 28 XXXXXXXXXXXXXXX | 37 XXXXXXXXXXXXXXX | 45 XXXXXXXXXXXXXXX |
| 2 C:1000:40:200       | 11              | 20 XXXXXXXXXXXXXXX | 29 XXXXXXXXXXXXXXX | 38 XXXXXXXXXXXXXXX | 46 XXXXXXXXXXXXXXX |
| 3 SOUT                | 12              | 21 XXXXXXXXXXXXXXX | 30 XXXXXXXXXXXXXXX | 39 XXXXXXXXXXXXXXX | 47 XXXXXXXXXXXXXXX |
| 4 PRT:FLOW=:OUT       | 13              | 22 XXXXXXXXXXXXXXX | 31 XXXXXXXXXXXXXXX | 40 XXXXXXXXXXXXXXX | 48 XXXXXXXXXXXXXXX |
| 5 PRT:TCP=:INP        | 14              | 23 XXXXXXXXXXXXXXX | 32 XXXXXXXXXXXXXXX | 41 XXXXXXXXXXXXXXX | 49 XXXXXXXXXXXXXXX |
| 6 END                 | 15              | 24 XXXXXXXXXXXXXXX | 33 XXXXXXXXXXXXXXX | 42 XXXXXXXXXXXXXXX | 50 XXXXXXXXXXXXXXX |
| 7                     | 16 XXXXXXXXXXXXXXX | 25 XXXXXXXXXXXXXXX | 34 XXXXXXXXXXXXXXX | 43 XXXXXXXXXXXXXXX | ( ONLY STEPS 1-15  |
| 8                     | 17 XXXXXXXXXXXXXXX | 26 XXXXXXXXXXXXXXX | 35 XXXXXXXXXXXXXXX | 44 XXXXXXXXXXXXXXX |  CURRENTLY ACTIVE ) |
|=====================================================================================================================|
| ** OUTPUT DATA                          | # OF FUNCTION STEPS     9. |      * SIGNAL VALUES *      |     * STORAGE REGISTERS * |
|                                         | INPT CMPNT REL EQU #    1. | PRESENT INPUT    0.425E+02 | 1   0.000E+00 6   0.000E+00 |
| ** BENCHMARKS (SIGNAL VALUES)           | INPT CMPNT STRM #       1. | LAST INPUT       0.425E+02 | 2   0.000E+00 7   0.000E+00 |
|          MIN         NOM         MAX   | OUTPT CMPNT REL EQU #   1. | PRESENT OUTPUT   0.500E+02 | 3   0.000E+00 8   0.000E+00 |
|INP   0.368E+02    0.414E+02    0.750E+02 | OUTPT CMPNT STRM #     99. | LAST OUTPUT      0.500E+02 | 4   0.000E+00 9   0.000E+00 |
|OUT   0.500E+02    0.500E+02    0.500E+02 | TOTAL # OF LOOPS      120. |                            | 5   0.000E+00               |
|_________________________________________|_________________________|____________________________|_____________________________|
```

Figure A-5. Sample problem 1: CNTRLLR edit screen.

A command list corresponding to the above edit session necessary for users operating on a graphics-only terminal is shown below. Data base management on a graphics terminal is possible only in a line editing mode. While in this mode, the user is continually prompted for necessary inputs. This is an abbreviated command list, i.e., the command prompts are omitted, and is intended to be referenced during an actual edit session.

```
CASEA_ATC > ED;PUMP     (enter line edit mode)

     13;50              (set flow rate)
     15;20              (set outlet pressure)
      1                 (page down in display)
     37;8               (set fluid code)
      0                 (exit)


CASEA_ATC > ED;TBUS     (enter line edit mode)

     14;35              (set control fluid temperature)
      0                 (exit)


CASEA_ATC > ED;CP       (enter line edit mode)

     13;1000            (set thermal load)
      0                 (exit)


CASEA_ATC > ED;CNTRLLR  (enter line edit mode)

     13;4               (active at the end of each time step)
     14;2               (input from PRO array)
     15;2               (location 2 in PRO - temperature)
     16;3               (output in CON array)
     17;10              (location 10 in CON - flow rate)
     18;PC:70:5:3       (see CNTRLLR description in section 10
     19;C:1000:40:200    for complete command list)
     20;SOUT
     21;PRT:FLOW=:OUT
     22;PRT:TCP=:INP
     23;END
      0                       (exit)
```

A.2.3 Solution Control and Execution

Before the solution is initiated, the user must verify that system level control parameters such as start time, stop time, time step, etc., are appropriate for the simulation to be performed. This system level information is contained in the CONTROL data base and is accessed in the same manner as the component data bases (section A.2.2). As above, the following procedures are for users operating on text terminals. Graphics terminal users should find no difficulty in adapting these steps for use in the line edit mode after a brief review of the previous section.

```
CASEA_ATC > ED;CONTROL
```

Using the arrow keys, move the cursor as necessary to set the start time to 0.0, the stop time to 3.0 h, and the time step to 0.01667 h (1.0 minute). The remaining parameters are discussed in section 4.1. The CONTROL edit screen is shown in figure A-6.

```
CASE NAME: ATC                CONTROL EDIT SCREEN          LAST UPDATED: 931130
                                                         RECORD NUMBER: 000111
| TIME UNIT FLAG (1=Hr,2=Min,3=Sec)    1 | PRESS SOLTN,0=Fdbk 1=Mtrx        1. |
| START TIME (Hr, Min or Sec)      0.000 | INITIAL PRESSURES,   Psia    14.700 |
| STOP TIME (Hr, Min or Sec)       3.000 | FLOW DMP FAC:M=G*dP,0<x<1 0.000E+00 |
| TIME STEP (Hr,Min or Sec)      0.01667 | SPECIFIED PROPS CONSTIT #        0. |
| OUTPUT INTERVAL(H,M or S)      1.00000 | SPECIFIED CP,    Btu/Lb/F 0.000E+00 |
| PLOT TIME UNITS (1=Hr,2=Min,3=Sec)  1. | SPECIFIED DENSITY, Lb/Ft3 0.000E+00 |
| CONVERGENCE CRITERION, 0<x<1 0.100E-04 | SPECIFIED VISC,   Lb/Ft/H 0.000E+00 |
| MAX SOLUTION ITERATIONS             75 | SPECIFIED CV,    Btu/Ft/H 0.000E+00 |
| NUMBER OF CONSTITUENTS, x<50         9 |
```

Figure A-6. Sample problem 1: CONTROL edit screen.

The final step before initiating a solution regards the preparation of any desired output. The IPU provides an efficient method of selecting simulation parameters to be stored in a data base (located in the user's directory) for later analysis. This parameter list is stored in the PLOT data base, which is accessed using the previously outlined techniques applicable to the CONTROL or component data bases. To realize the full potential of the IPU, the user must have access to a text terminal. For the purposes of this example, create a data base (plotset) named ATCPLOT which will allow tracking of the fluid temperatures between the three components as a function of time. Enter the PLOT data base using the `ED` command:

```
CASEA_ATC > ED;PLOT
```

The primary edit screen, as shown in figure A-7, is used for naming a plotset (limited to eight characters), specifying the OPS block in which it is active, and activating/deactivating the plotset. Reference sections 4.3 and 5.4 for additional information concerning the IPU. Begin by entering the name of the plotset: ATCPLOT. Next indicate that this plotset operates in conjunction with the output OPS logic (OPS 4) by entering a 4 in the "OPS#" column, and set the activation flag to 1 (on).

```
CASE NAME: ATC                  PLOT EDIT SCREEN                LAST UPDATED: 941005
                                                               RECORD NUMBER: 000010
|-PLOTSET NAME-OPS # -SAMPLE--PLOT--AUTOPLOT--ARCHIVE--# OF REPETITIONS--^L TO OPEN-|
|  1. ATCPLOT    4     001      1       1        1            01           00119    |
|  2.            0     000      0       0        0            00           00000    |
|  3.            0     000      0       0        0            00           00000    |
|  4.            0     000      0       0        0            00           00000    |
|  5.            0     000      0       0        0            00           00000    |
|  6.            0     000      0       0        0            00           00000    |
|  7.            0     000      0       0        0            00           00000    |
|  8.            0     000      0       0        0            00           00000    |
|  9.            0     000      0       0        0            00           00000    |
| 10.            0     000      0       0        0            00           00000    |
| 11.            0     000      0       0        0            00           00000    |
| 12.            0     000      0       0        0            00           00000    |
| 13.            0     000      0       0        0            00           00000    |
| 14.            0     000      0       0        0            00           00000    |
| 15.            0     000      0       0        0            00           00000    |
| 16.            0     000      0       0        0            00           00000    |
| 17.            0     000      0       0        0            00           00000    |
| 18.            0     000      0       0        0            00           00000    |
| 19.            0     000      0       0        0            00           00000    |
| 20.            0     000      0       0        0            00           00000    |
```

Figure A-7. Sample problem 1: Primary PLOT edit screen.

In order to define the actual data to be used in this plotset, an additional edit screen is necessary. This screen is created by positioning the cursor in the eighth column (^L TO OPEN) and entering CNTRL-L. The Plotset Definition Screen will appear as shown in figure A-8. As an example, let the first item in the data base be time. Thus, enter “TIME” in the column labeled code. Next, skip the columns labeled COMPONENT NAME, LOCATION, and STREAM#. Each CODE type requires entries to fully define the desired parameter, as described in section 4.3. Since time is independent of the component names, array locations, and component stream numbers, these columns require no entries. Finally, enter an appropriate title in the last data field, namely, “TIME, HRS”. Hit the TAB key to return the cursor to the first column. Create a line for the second item in the parameter list by hitting the PF3 key (CNTRL-B). Let this item be the flow rate of the PUMP component. Mass flow rate is tracked in location 1 of the C array for each stream of every component (see section 7.2). Therefore, enter “C” in the CODE column. Note that alphanumeric entries must be left justified in a data field. Next, enter “P-1” (the name of the PUMP component) in the second column. Finally, indicate the correct array location and stream number by entering a “1” in the third column and a “2” in the fourth column (the outlet side of the PUMP component is stream 2). Numeric entries should be right justified in a data field. Lastly, enter a logical title for this item: “PUMP FLOW RATE, LB-H2O/HR”. Continue in this manner to create items for the outlet temperatures of the thermal bus and cold plate components (location 2 in the “PRO” array, stream 2). When all five items have been entered, enter CNTRL Z to save the changes and return to the main PLOT edit screen. Just as with other data base edit sessions, the user may exit without saving any changes by entering CNTRL X (escape). To return to the command prompt, enter CNTRL Z.

```
                          PLOTSET DEFINITION

 CODE   COMPONENT NAME  LOCATION  STREAM#                  TITLE
 ----   --------------  --------  -------     --------------------------------
 TIME                     0000       0        TIME, HRS
 C         P-1            0001       2        PUMP FLOW RATE, LB-H2O/HR
 PRO       PAYLOAD        0002       2        TBUS OUTLET TEMP, F
 PRO       LOAD-1         0002       2        CP OUTLET TEMP,   F
 CON       LOAD-1         0028       0        EQUIPMENT TEMP,   F
```

Figure A-8. Sample problem 1: Secondary PLOT edit screen.

At this point, the model is ready for solution execution. To initiate the solution process, enter the SOLVE command:

```
CASEA_ATC > SOLVE
```

Once the command has been issued, the solution routine will take control and the user has no method of stopping the simulation except by using the CNTRL Y interrupt feature of DEC VMS operating system. This will return the user to the DCL level. The user is cautioned to never interrupt the solution until the pseudo-compute sequence construction is complete (see the text messages below). The simulation may be resumed by entering CONTinue. If any other DCL command is executed, the solution process will be canceled. In this case, the user must reexecute the CASE/A program, reload the case using the `LOADCASE` command and reissue the `SOLVE` command. As the solution proceeds, messages will be sent to the screen informing the user of the simulation status:

```
CASEA_ATC > SOLVE
READING SIMULATION CONTROL DATA....
CHECKING FOR ACTIVE PLOTSETS...
READING COMPONENT DATA BASE PARAMETERS...
CONSTRUCTING PSEUDO-COMPUTE SEQUENCE AND FLOW MAP...
PERFORMING COMPONENT INITIALIZATION OPERATIONS...
INITIATING SIMULATION SOLUTION...

**** TIME=  0.0000E+00  RELX=  0.1000E-04  RXCC=  0.0000E+00  LOOPS=  0
```

```
**** TIME=  0.1667E-01  RELX=  0.1000E-04  RXCC=  0.3452E-05  LOOPS=  7
**** TIME=  0.3334E-01  RELX=  0.1000E-04  RXCC=  0.3018E-05  LOOPS=  4
            .
            .
            .
**** TIME=  0.3000E+01  RELX=  0.1000E-04  RXCC=  0.2997E-05  LOOPS=  3
**** SIMULATION COMPLETED:  PERFORMING WRAP-UP...

CASEA_ATC >
```

A.2.4 Output Management

Upon completion of the simulation, the user may examine the results at each output interval (as specified in the CONTROL data base) by accessing the ATC.LPP file using the `EDT` command or directly through the VAX/VMS text editor from the DCL level. Information concerning this file may be found in section 5.6. The data saved via the Integrated Plot Utility may be accessed by executing the `IPU` command and entering ATCPLOT when prompted for a data base name:

```
CASEA_ATC > IPU
ENTER DATA BASE NAME: ATCPLOT
```

An error message will inform the user if there was a problem loading the data base. Otherwise, the `FULL` command must be issued to activate the appropriate data base records:

```
IPU_ATCPLOT > FULL
TOTAL RECORDS IN TABLE :     180
```

The `LIST` command will remind the user of the parameters selected during the set up of the plotset:

```
IPU_ATCPLOT > LIST
         1 TIME, HRS                          0.00000000E+00
         2 PUMP FLOW RATE, LB-H2O/HR          0.00000000E+00
         3 TBUS OUTLET TEMP, F                0.00000000E+00
         4 CP OUTLET TEMP,   F                0.00000000E+00
         5 EQUIPMENT TEMP,   F                0.00000000E+00
         6 MOD DATE                           0
         7 SECURITY
```

Note that this listing corresponds to the data record number 0, i.e., the data stored in the data base at time zero. If the `LIST 1` command was issued, the values shown would reflect the data contained in record 1 corresponding to the first time step, 0.01667 h (elapsed simulation time). As can be seen, the 180 records correspond to the 180 time steps (3 h at 1 min time steps) of the simulation. The record at time zero is not counted as one of the records in the table. Also, the modification date (MOD DATE) and security password (SECURITY) are included automatically. The security item is intended to prevent inadvertent errors rather than malicious data overwrites and is currently disabled. A variety of IPU commands are available for data analysis and/or examination. These commands are discussed in section 5.4. The following steps illustrate the procedure for obtaining a hard copy plot of the three temperatures versus time. Note that an HP laserjet printer must be supported on the network. Begin by setting the terminal type to send data to a printer rather than the screen (see discussion of the `TERM` command in section 6.1):

```
IPU_ATCPLOT > TERM;1
```

To obtain a plot of items 3, 4, and 5 versus 1 (time), issue the `PLOT0` command as follows:

```
IPU_ATCPLOT > PLOT0;1;3;4;5
```

The user will be notified with the appropriate system message indicating the status of the printing job. The first argument of the `PLOT0` command becomes the independent variable on the graph. There are many options regarding the format of the graph including limits on the magnitude of the independent or dependent variables, headings and labels, logos, etc. The user should see section 5.4 for further details. The plot obtained from the execution of the above command is shown in figure A-9.

To close the current work session, save the case in the current configuration, return to the VAX/VMS environment, and execute the `EXIT` command.

```
IPU_ATCPLOT > EXIT
```

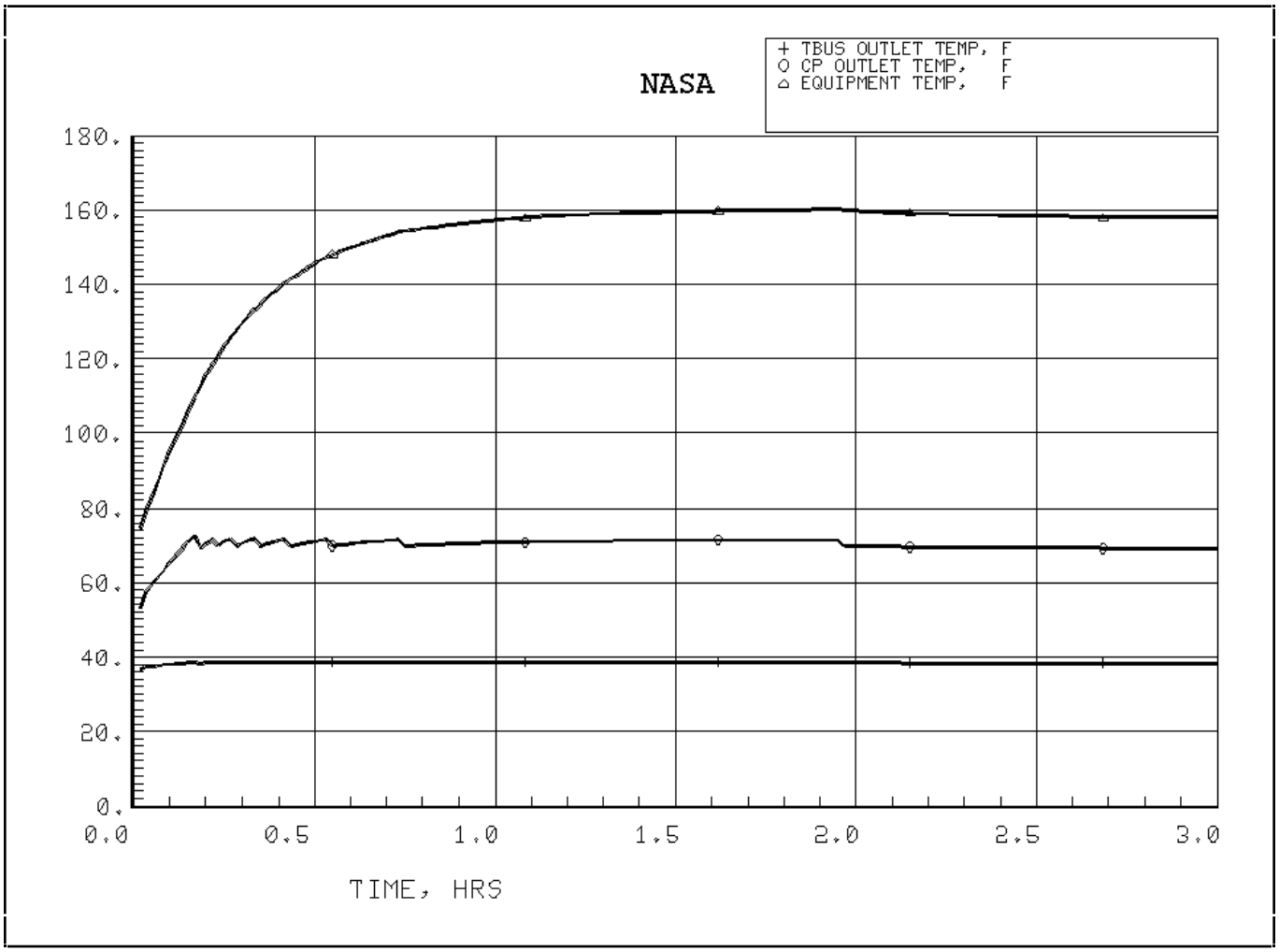


Figure A-9. Sample problem 1: Temperatures versus time.

A.3 SAMPLE PROBLEM 2

The following sample problem illustrates the implementation of user-defined operations logic (OPS logic) and the constants data base (USERCON). A CASE/A model is constructed to simulate the operation of a water conditioning system to assess the energy requirements of a centralized hot water supply system for the space station HAB module. The OPS logic was necessary to model the "on demand" operation of the hot water supply system. The following sections provide a complete problem description and an abbreviated outline of the model construction. Consult sample problem 1 for additional information regarding subsystem assembly or data base management, if necessary.

A.3.1 Problem Description

The HAB module hot water supply is required to provide hot water for handwash, shower, dishwash, and laundry facilities. The water is supplied to these facilities from a centralized storage tank. The tank water is continuously circulated at a rate of 15 lbm/h through a heater to maintain an outlet temperature of 145 °F. The storage tank initially contains 113 lbm of water and is resupplied at a rate of 15 lbm/h when water is being drawn from the tank. Rack plumbing for the system consists of pipelines from the stand-off line at four separate use points with an additional line running from the storage tank loop. All plumbing lines are assumed to be 1/16-inch thick stainless steel pipe of 1/2-inch O.D. The hot water requirements of the HAB module handwash, shower, dishwash, and laundry over a 24 hour period are defined as follows:

Handwash: 1 lbm of 110 °F water at 3 lbm/min; 1 handwash for each crew member at 5-min intervals; repeat cycle at 4-h intervals; 4 cycles in a 24-h period.

Shower: 8 lbm of 110 °F water at 3 lbm/min; 1 shower for each crew member every 24 h; showers are 45 min apart.

Dishwash: 48 lbm of 140 °F water at 3 lbm/min; 1 dishwash in a 24-h period.

Laundry: 50 lbm of 140 °F water at 3 lbm/min; 2 washes in a 24-h period at 8-h intervals.

The simulation should begin with all use points requiring hot water and an 8 member crew.

A.3.2 Model Development

The subsystem schematic of the water conditioning model (WCM) is shown in figure A-10. The edit screens reflecting the desired performance data are shown in figures A-11 through A-31. The solution control data, user-defined constants data, and the plotset data are shown in figures A-32 through A-35. The task of locating the components and making the appropriate connections, as well as modifying the data bases, is left to the user. See sample problem 1 for assistance.

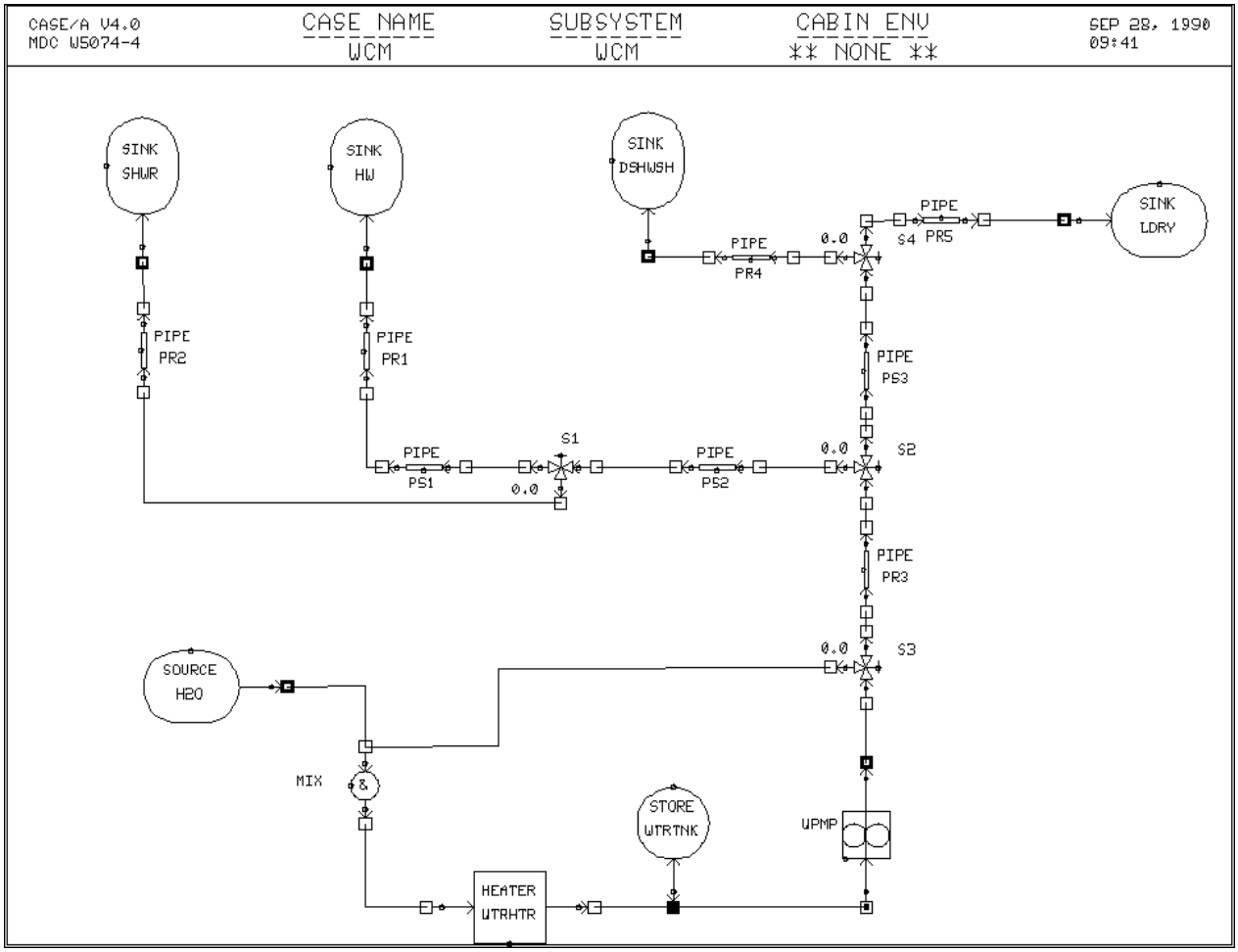


Figure A-10. Sample problem 2: WCM schematic.

```
CASE NAME: WCM                SINK EDIT SCREEN              LAST UPDATED: 890624
SUBSYSTEM: WCM                COMPONENT: HW                RECORD NUMBER: 000058
_____________________________________________________________________________
|       ** GENERAL INPUT DATA **                                            |
| 10 BACK PRESSURE     (PSIA)   14.70 |  * RESOURCE TRACKING *               |
|                                     | 7 POWER   (WATTS)   0.00             |
|                                     | 8 WEIGHT  (LBM)     0.00             |
|                                     | 9 VOLUME  (FT^3)    0.00             |
|                                     |                                      |
|_____________________________________|______________________________________|
| ** BENCHMARK DATA                                  | ** OUTPUT DATA          |
|                       MIN        NOM        MAX   |13 ACCM MASS,Lb      8.28 |
| INLET FLOW (LB/H)  0.00E+00    8.2600016  0.18E+03|14 FLOW IN,Lb/H  0.000E+00|
| INLET TEMP  (DeF) 125.13E+03   144.09372 144.82147|15 TEMP IN,DegF 144.430420|
| INLET PRESS (PSIA) 14.700000  14.699961  14.700000|16 PRES IN,Psia   14.7    |
|____________________________________________________|_________________________|
_____________________________________________________________________________
```

Figure A-11. Sample problem 2: SINK edit screen (HW).

```
CASE NAME: WCM                SINK EDIT SCREEN              LAST UPDATED: 890624
SUBSYSTEM: WCM                COMPONENT: SHW               RECORD NUMBER: 000058
_____________________________________________________________________________
|       ** GENERAL INPUT DATA **                                            |
| 10 BACK PRESSURE     (PSIA)   14.70 |  * RESOURCE TRACKING *               |
|                                     | 7 POWER   (WATTS)   0.00             |
|                                     | 8 WEIGHT  (LBM)     0.00             |
|                                     | 9 VOLUME  (FT^3)    0.00             |
|                                     |                                      |
|_____________________________________|______________________________________|
|           ** BENCHMARK DATA **                    |    ** OUTPUT DATA **    |
|                       MIN        NOM        MAX   |13 ACCM MASS,Lb      16.2 |
| INLET FLOW  (LB/H)  0.00E+00  16.199995   0.18E+03|14 FLOW IN,Lb/H  0.000E+00|
| INLET TEMP  (DeF) 125.13E+03  144.09372  144.82147|15 TEMP IN,DegF 144.430420|
| INLET PRESS (PSIA) 14.700000  14.699961  14.700000|16 PRES IN,Psia   14.7    |
|_____________________________________________________|_________________________|
_____________________________________________________________________________
```

Figure A-12. Sample problem 2: SINK edit screen (SHWR).

```
CASE NAME: WCM                SINK EDIT SCREEN              LAST UPDATED: 890624
SUBSYSTEM: WCM                COMPONENT: LDRY              RECORD NUMBER: 000058
_____________________________________________________________________________
|       ** GENERAL INPUT DATA **                                            |
| 10 BACK PRESSURE     (PSIA)   14.70 |  * RESOURCE TRACKING *               |
|                                     | 7 POWER   (WATTS)   0.00             |
|                                     | 8 WEIGHT  (LBM)     0.00             |
|                                     | 9 VOLUME  (FT^3)    0.00             |
|                                     |                                      |
|_____________________________________|______________________________________|
|           ** BENCHMARK DATA **                    |    ** OUTPUT DATA **    |
|                       MIN        NOM        MAX   |13 ACCM MASS,Lb      49.7 |
| INLET FLOW  (LB/H)  0.00E+00  49.880047   0.18E+03|14 FLOW IN,Lb/H  0.000E+00|
| INLET TEMP  (DeF)  143.66794  144.67255  144.88977|15 TEMP IN,DegF   145.   |
| INLET PRESS (PSIA) 14.700000  14.699961  14.700000|16 PRES IN,Psia   14.7    |
|_____________________________________________________|_________________________|
_____________________________________________________________________________
```

Figure A-13. Sample problem 2: SINK edit screen (DSHWSH).

```
CASE NAME: WCM                SINK EDIT SCREEN              LAST UPDATED: 890624
SUBSYSTEM: WCM                COMPONENT: DSHWSH            RECORD NUMBER: 000058
_____________________________________________________________________________
|       ** GENERAL INPUT DATA **                                            |
| 10 BACK PRESSURE     (PSIA)   14.70 |  * RESOURCE TRACKING *               |
|                                     | 7 POWER   (WATTS)   0.00             |
|                                     | 8 WEIGHT  (LBM)     0.00             |
|                                     | 9 VOLUME  (FT^3)    0.00             |
|                                     |                                      |
|_____________________________________|______________________________________|
|           ** BENCHMARK DATA **                    |    ** OUTPUT DATA **    |
|                       MIN        NOM        MAX   |13 ACCM MASS,Lb      47.9 |
| INLET FLOW  (LB/H)  0.00E+00  47.880047   0.18E+03|14 FLOW IN,Lb/H  0.000E+00|
| INLET TEMP  (DeF)  144.06839  144.09372  144.73219|15 TEMP IN,DegF 144.465942|
| INLET PRESS (PSIA) 14.700000  14.699961  14.700000|16 PRES IN,Psia   14.7    |
|_____________________________________________________|_________________________|
_____________________________________________________________________________
```

Figure A-14. Sample problem 2: SINK edit screen (LDRY).

```
CASE NAME: WCM                                   PIPE EDIT SCREEN                                LAST UPDATED: 890627
SUBSYSTEM: WCM          COMPONENT: PR1                                                        RECORD NUMBER: 000031

 ** GENERAL INPUT DATA **                      |** RESOURCE TRACKING **|        ** THERMAL INTERFACE DATA **
 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)    1.0  | 9 VOLUME  (FT^3) 0.00 | 14 SHEL-ENV CONVECTIVE H*A (BTU/HR/F) 0.932
 11 HYDRAULIC EQUIVALENT LENGTH     (FT) 7.12  | 8 WEIGHT  (LBM)  0.00 | 15 SHEL-ENV RADIATIVE F*A*E (FT^2)   0.210
 12 HYDRAULIC EQUIVALENT DIAMETER  (IN)  0.37  | 7 POWER  (WATTS) 0.00 | 16 SHEL-ENV CONDUCTIVE KA/X (BTU/HR/F) 0.249
 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)    0.00  |                       | 17 MASS-SHEL CONDUCT KA/X (BTU/HR/F)  4.04
                                               |** SOLUTION CONTROL ** | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)0.90000
 20 INITIAL MASS TEMPERATURE      (F)  145.00  | 22 RELAXATION 0.00001 | 19 WALL-MASS CONDUCT KA/X (BTU/HR300.00
 21 THERMAL CAPACITANCE       (BTU/F)   0.240  | 23 MAX LOOP CNT 100.0 |
** OUTPUT DATA **                             | ** BENCHMARK DATA **              MIN         NOM         MAX
       25 PRESENT MASS TEMPERATURE     (F)   81.55   |    MASS TEMPERATURE     (F)  82.     0.11E+03    0.14E+03
       26 PRESENT SHELL TEMPERATURE    (F)   79.86   |    SHELL TEMPERATURE    (F)  80.     0.10E+03    0.13E+03
       27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00   |    HEAT LOSS QDOT  (BTU/HR)  0.00E+00   8.3       0.18E+03
```

Figure A-15. Sample problem 2: PIPE edit screen (PR1).

```
CASE NAME: WCM                                   PIPE EDIT SCREEN                                LAST UPDATED: 890627
SUBSYSTEM: WCM          COMPONENT: PS1                                                        RECORD NUMBER: 000032

 ** INPUT DATA                                   * RESOURCE TRACKING *            * THERMAL INTERFACE DATA *
 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)      1.0  | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.500
 11 HYDRAULIC EQUIVALENT LENGTH      (FT)   3.82  | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.113
 12 HYDRAULIC EQUIVALENT DIAMETER    (IN)   0.37  | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   3.900
 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)       0.00  |                          | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)   2.17
                                                  |  *SUB SOLUTION CONTROL*  | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000
 20 INITIAL MASS TEMPERATURE          (F)  145.00  | 22 RELAXATION    0.00001 | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00
 21 THERMAL CAPACITANCE           (BTU/F)   0.127  | 23 MAX LOOP CNT    100.0 |
 ** OUTPUT DATA                                   | ** BENCHMARK DATA              MIN         NOM         MAX
            25 PRESENT MASS TEMPERATURE     (F)   75.09    |    MASS TEMPERATURE     (F)  75.         98.        0.14E+03
            26 PRESENT SHELL TEMPERATURE    (F)   75.03    |    SHELL TEMPERATURE    (F)  75.         82.         97.
            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00    |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00     8.3        0.18E+03
```

Figure A-16. Sample problem 2: PIPE edit screen (PS1).

```
CASE NAME: WCM                                   PIPE EDIT SCREEN                                LAST UPDATED: 890627
SUBSYSTEM: WCM          COMPONENT: PR2                                                        RECORD NUMBER: 000033

 ** INPUT DATA                                   * RESOURCE TRACKING *            * THERMAL INTERFACE DATA *
 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)      1.0  | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.932
 11 HYDRAULIC EQUIVALENT LENGTH      (FT)   7.12  | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.210
 12 HYDRAULIC EQUIVALENT DIAMETER    (IN)   0.37  | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   0.250
 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)       0.00  |                          | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)   4.04
                                                  |  *SUB SOLUTION CONTROL*  | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000
 20 INITIAL MASS TEMPERATURE          (F)  145.00  | 22 RELAXATION    0.00001 | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00
 21 THERMAL CAPACITANCE           (BTU/F)   0.240  | 23 MAX LOOP CNT    100.0 |
 ** OUTPUT DATA                                   | ** BENCHMARK DATA              MIN         NOM         MAX
            25 PRESENT MASS TEMPERATURE     (F)  102.65    |    MASS TEMPERATURE     (F)  76.       0.10E+03    0.14E+03
            26 PRESENT SHELL TEMPERATURE    (F)   95.47    |    SHELL TEMPERATURE    (F)  76.        93.        0.13E+03
            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00    |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00     16.        0.18E+03
```

Figure A-17. Sample problem 2: PIPE edit screen (PR2).

```
CASE NAME: WCM                                   PIPE EDIT SCREEN                                LAST UPDATED: 890627
SUBSYSTEM: WCM          COMPONENT: PS2                                                        RECORD NUMBER: 000034

 ** INPUT DATA                                   * RESOURCE TRACKING *            * THERMAL INTERFACE DATA *
 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)      1.0  | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.400
 11 HYDRAULIC EQUIVALENT LENGTH      (FT)   3.05  | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.089
 12 HYDRAULIC EQUIVALENT DIAMETER    (IN)   0.37  | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   3.900
 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)       0.00  |                          | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)   1.74
                                                  |  *SUB SOLUTION CONTROL*  | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000
 20 INITIAL MASS TEMPERATURE          (F)  145.00  | 22 RELAXATION    0.00001 | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00
 21 THERMAL CAPACITANCE           (BTU/F)   0.101  | 23 MAX LOOP CNT    100.0 |
 ** OUTPUT DATA                                   | ** BENCHMARK DATA              MIN         NOM         MAX
            25 PRESENT MASS TEMPERATURE     (F)   80.18    |    MASS TEMPERATURE     (F)  80.       0.11E+03    0.14E+03
            26 PRESENT SHELL TEMPERATURE    (F)   76.47    |    SHELL TEMPERATURE    (F)  76.        85.        95.
            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00    |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00     24.        0.36E+03
```

Figure A-18. Sample problem 2: PIPE edit screen (PS2).

```
CASE NAME: WCM                                        PIPE EDIT SCREEN                                  LAST UPDATED: 890627
SUBSYSTEM: WCM           COMPONENT: PS3                                                              RECORD NUMBER: 000035
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                        * RESOURCE TRACKING *              * THERMAL INTERFACE DATA *            |
| 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)          1.0   | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.458  |
| 11 HYDRAULIC EQUIVALENT LENGTH       (FT)    3.50   | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.103  |
| 12 HYDRAULIC EQUIVALENT DIAMETER     (IN)    0.37   | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   3.900  |
| 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)          0.00   |                              | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)    1.99  |
|                                                      |  *SUB SOLUTION CONTROL*      | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000  |
| 20 INITIAL MASS TEMPERATURE           (F)  145.00    | 22 RELAXATION    0.00001     | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00  |
| 21 THERMAL CAPACITANCE            (BTU/F)   0.116    | 23 MAX LOOP CNT    100.0     |                                                      |
|_____________________________________________________|____________________________|_____________________________________________|
| ** OUTPUT DATA                                       | ** BENCHMARK DATA              MIN         NOM         MAX          |
|            25 PRESENT MASS TEMPERATURE     (F)   75.00 |    MASS TEMPERATURE      (F)  75.        97.       0.14E+03          |
|            26 PRESENT SHELL TEMPERATURE    (F)   75.00 |    SHELL TEMPERATURE     (F)  75.        82.      96.                 |
|            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00 |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00    98.       0.36E+03          |
|_____________________________________________________|____________________________|_____________________________________________|
```

Figure A-19. Sample problem 2: PIPE edit screen (PS3).

```
CASE NAME: WCM                                        PIPE EDIT SCREEN                                  LAST UPDATED: 890627
SUBSYSTEM: WCM           COMPONENT: PR4                                                              RECORD NUMBER: 000036
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                        * RESOURCE TRACKING *              * THERMAL INTERFACE DATA *            |
| 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)          1.0   | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.932  |
| 11 HYDRAULIC EQUIVALENT LENGTH       (FT)    7.12   | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.210  |
| 12 HYDRAULIC EQUIVALENT DIAMETER     (IN)    0.37   | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   0.250  |
| 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)          0.00   |                              | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)    4.04  |
|                                                      |  *SUB SOLUTION CONTROL*      | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000  |
| 20 INITIAL MASS TEMPERATURE           (F)  145.00    | 22 RELAXATION    0.00001     | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00  |
| 21 THERMAL CAPACITANCE            (BTU/F)   0.240    | 23 MAX LOOP CNT    100.0     |                                                      |
|_____________________________________________________|____________________________|_____________________________________________|
| ** OUTPUT DATA                                       | ** BENCHMARK DATA              MIN         NOM         MAX          |
|            25 PRESENT MASS TEMPERATURE     (F)   76.22 |    MASS TEMPERATURE      (F)  76.        0.11E+03    0.14E+03          |
|            26 PRESENT SHELL TEMPERATURE    (F)   75.90 |    SHELL TEMPERATURE     (F)  76.        97.         0.13E+03          |
|            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00 |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00    48.         0.18E+03          |
|_____________________________________________________|____________________________|_____________________________________________|
```

Figure A-20. Sample problem 2: PIPE edit screen (PR4).

```
CASE NAME: WCM                                        PIPE EDIT SCREEN                                  LAST UPDATED: 890627
SUBSYSTEM: WCM           COMPONENT: PR5                                                              RECORD NUMBER: 000037
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                        * RESOURCE TRACKING *              * THERMAL INTERFACE DATA *            |
| 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)          1.0   | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.932  |
| 11 HYDRAULIC EQUIVALENT LENGTH       (FT)    7.12   | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.210  |
| 12 HYDRAULIC EQUIVALENT DIAMETER     (IN)    0.37   | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   0.250  |
| 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)          0.00   |                              | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)    4.04  |
|                                                      |  *SUB SOLUTION CONTROL*      | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000  |
| 20 INITIAL MASS TEMPERATURE           (F)  145.00    | 22 RELAXATION    0.00001     | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00  |
| 21 THERMAL CAPACITANCE            (BTU/F)   0.240    | 23 MAX LOOP CNT    100.0     |                                                      |
|_____________________________________________________|____________________________|_____________________________________________|
| ** OUTPUT DATA                                       | ** BENCHMARK DATA              MIN         NOM         MAX          |
|            25 PRESENT MASS TEMPERATURE     (F)   76.28 |    MASS TEMPERATURE      (F)  76.        0.11E+03    0.14E+03          |
|            26 PRESENT SHELL TEMPERATURE    (F)   75.95 |    SHELL TEMPERATURE     (F)  76.        98.         0.13E+03          |
|            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00 |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00    50.         0.18E+03          |
|_____________________________________________________|____________________________|_____________________________________________|
```

Figure A-21. Sample problem 2: PIPE edit screen (PR5).

```
CASE NAME: WCM                                        PIPE EDIT SCREEN                                  LAST UPDATED: 890627
SUBSYSTEM: WCM           COMPONENT: PR3                                                              RECORD NUMBER: 000038
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                        * RESOURCE TRACKING *              * THERMAL INTERFACE DATA *            |
| 10 FLOW DIRECTION (0=?, 1=1-2, 2=2-1)          1.0   | 9 VOLUME  (FT^3)    0.00 | 14 SHEL-ENV CONVECTIVE H*A   (BTU/HR/F)   0.932  |
| 11 HYDRAULIC EQUIVALENT LENGTH       (FT)    7.12   | 8 WEIGHT   (LBM)    0.00 | 15 SHEL-ENV RADIATIVE F*A*E      (FT^2)   0.210  |
| 12 HYDRAULIC EQUIVALENT DIAMETER     (IN)    0.37   | 7 POWER  (WATTS)    0.00 | 16 SHEL-ENV CONDUCTIVE KA/X  (BTU/HR/F)   0.250  |
| 13 HYDRAULIC K FACTOR (K=Dp(X/M)**2)          0.00   |                              | 17 MASS-SHEL CONDUCT KA/X    (BTU/HR/F)    4.04  |
|                                                      |  *SUB SOLUTION CONTROL*      | 18 FLUID-WALL HEAT TRANS EFF (0<X<1)    0.90000  |
| 20 INITIAL MASS TEMPERATURE           (F)  145.00    | 22 RELAXATION    0.00001     | 19 WALL-MASS CONDUCT KA/X    (BTU/HR/F)  300.00  |
| 21 THERMAL CAPACITANCE            (BTU/F)   0.240    | 23 MAX LOOP CNT    100.0     |                                                      |
|_____________________________________________________|____________________________|_____________________________________________|
| ** OUTPUT DATA                                       | ** BENCHMARK DATA              MIN         NOM         MAX          |
|            25 PRESENT MASS TEMPERATURE     (F)  103.00 |    MASS TEMPERATURE      (F)  98.        0.13E+03    0.14E+03          |
|            26 PRESENT SHELL TEMPERATURE    (F)   95.74 |    SHELL TEMPERATURE     (F)  92.        0.11E+03    0.13E+03          |
|            27 PRESENT HEAT LOSS QDOT  (BTU/HR)    0.00 |    HEAT LOSS QDOT   (BTU/HR) 0.00E+00    0.12E+03    0.72E+03          |
|_____________________________________________________|____________________________|_____________________________________________|
```

Figure A-22. Sample problem 2: PIPE edit screen (PR3).

```
CASE NAME: WCM                                        SPLIT EDIT SCREEN                                  LAST UPDATED: 890624
SUBSYSTEM: WCM           COMPONENT: S1                                                                  RECORD NUMBER: 000029
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                       * RESOURCE TRACKING *                    * THERMAL INTERFACE DATA *       |
| 10 NUMBER OF OUTLET STREAMS              2.0 | 7 POWER                  (WATTS)    0.00 | 17 ** OPEN INPUT LOCATION **       0.000 |
| 13 FLOW SOLUTION DAMPING CONSTNT 0.5000000 | 8 WEIGHT                   (LBM)    0.00 | 18 ** OPEN INPUT LOCATION **       0.000 |
| 11 SPLIT FRACTION OF OUTLT LEG 1 0.0000000 | 9 VOLUME                  (FT^3)    0.00 | 19 ** OPEN INPUT LOCATION **       0.000 |
| 12 SPLIT FRACTION OF OUTLT LEG 2 1.0000000 |                                        | 20 ** OPEN INPUT LOCATION **       0.000 |
| 14 HYDRAUL K FACT OF OUTLT LEG 1  0.00E+00 |                                        | 21 ** OPEN INPUT LOCATION **       0.000 |
| 15 HYDRAUL K FACT OF OUTLT LEG 2  0.00E+00 |                                        |                                          |
| ** BENCHMARK DATA                          |                                | ** OUTPUT DATA                           |
|                          MIN         NOM         MAX                |        26 OPERATION MODE FLAG           1.000                |
| INLET FLOW    (LBM/HR)     0.000     24.480     360.000              |        23 INLET FLOW     (LBM/HR)       0.000                |
| SPLIT PRESSURE  (PSIA)    14.700     14.711      14.800              |        24 SPLIT PRESSURE   (PSIA)      14.700                |
| FLOW TEMPERATURE   (F)   131.672    144.285     144.836              |        25 FLOW TEMPERATURE    (F)     144.518                |
|____________________________________________________________________|____________________________________________________________|
```

Figure A-23. Sample problem 2: SPLIT edit screen (S1).

```
CASE NAME: WCM                                        SPLIT EDIT SCREEN                                  LAST UPDATED: 890624
SUBSYSTEM: WCM           COMPONENT: S2                                                                  RECORD NUMBER: 000030
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                       * RESOURCE TRACKING *                    * THERMAL INTERFACE DATA *       |
| 10 NUMBER OF OUTLET STREAMS              2.0 | 7 POWER                  (WATTS)    0.00 | 17 ** OPEN INPUT LOCATION **       0.000 |
| 13 FLOW SOLUTION DAMPING CONSTNT 0.5000000 | 8 WEIGHT                   (LBM)    0.00 | 18 ** OPEN INPUT LOCATION **       0.000 |
| 11 SPLIT FRACTION OF OUTLT LEG 1 0.0000000 | 9 VOLUME                  (FT^3)    0.00 | 19 ** OPEN INPUT LOCATION **       0.000 |
| 12 SPLIT FRACTION OF OUTLT LEG 2 1.0000000 |                                        | 20 ** OPEN INPUT LOCATION **       0.000 |
| 14 HYDRAUL K FACT OF OUTLT LEG 1  0.00E+00 |                                        | 21 ** OPEN INPUT LOCATION **       0.000 |
| 15 HYDRAUL K FACT OF OUTLT LEG 2  0.00E+00 |                                        |                                          |
| ** BENCHMARK DATA                          |                                | ** OUTPUT DATA                           |
|                          MIN         NOM         MAX                |        26 OPERATION MODE FLAG           1.000                |
| INLET FLOW    (LBM/HR)     0.000    122.040     720.000              |        23 INLET FLOW     (LBM/HR)       0.000                |
| SPLIT PRESSURE  (PSIA)    14.700     14.751      14.869              |        24 SPLIT PRESSURE   (PSIA)      14.700                |
| FLOW TEMPERATURE   (F)   137.270    144.460     144.965              |        25 FLOW TEMPERATURE    (F)     144.518                |
|____________________________________________________________________|____________________________________________________________|
```

Figure A-24. Sample problem 2: SPLIT edit screen (S2).

```
CASE NAME: WCM                                        SPLIT EDIT SCREEN                                  LAST UPDATED: 890627
SUBSYSTEM: WCM           COMPONENT: S3                                                                  RECORD NUMBER: 000031
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                       * RESOURCE TRACKING *                    * THERMAL INTERFACE DATA *       |
| 10 NUMBER OF OUTLET STREAMS              2.0 | 7 POWER                  (WATTS)    0.00 | 17 ** OPEN INPUT LOCATION **       0.000 |
| 13 FLOW SOLUTION DAMPING CONSTNT 0.5000000 | 8 WEIGHT                   (LBM)    0.00 | 18 ** OPEN INPUT LOCATION **       0.000 |
| 11 SPLIT FRACTION OF OUTLT LEG 1 0.0000000 | 9 VOLUME                  (FT^3)    0.00 | 19 ** OPEN INPUT LOCATION **       0.000 |
| 12 SPLIT FRACTION OF OUTLT LEG 2 1.0000000 |                                        | 20 ** OPEN INPUT LOCATION **       0.000 |
| 14 HYDRAUL K FACT OF OUTLT LEG 1  0.00E+00 |                                        | 21 ** OPEN INPUT LOCATION **       0.000 |
| 15 HYDRAUL K FACT OF OUTLT LEG 2  0.00E+00 |                                        |                                          |
| ** BENCHMARK DATA                          |                                | ** OUTPUT DATA                           |
|                          MIN         NOM         MAX                |        26 OPERATION MODE FLAG           1.000                |
| INLET FLOW    (LBM/HR)    10.000    132.039     730.000              |        23 INLET FLOW     (LBM/HR)      10.000                |
| SPLIT PRESSURE  (PSIA)    15.000     15.032      15.303              |        24 SPLIT PRESSURE   (PSIA)      15.000                |
| FLOW TEMPERATURE   (F)   143.213    144.081     144.998              |        25 FLOW TEMPERATURE    (F)     143.308                |
|____________________________________________________________________|____________________________________________________________|
```

Figure A-25. Sample problem 2: SPLIT edit screen (S3).

```
CASE NAME: WCM                                        SPLIT EDIT SCREEN                                  LAST UPDATED: 890624
SUBSYSTEM: WCM           COMPONENT: S4                                                                  RECORD NUMBER: 000032
_____________________________________________________________________________________________________________________________
| ** INPUT DATA                                       * RESOURCE TRACKING *                    * THERMAL INTERFACE DATA *       |
| 10 NUMBER OF OUTLET STREAMS              2.0 | 7 POWER                  (WATTS)    0.00 | 17 ** OPEN INPUT LOCATION **       0.000 |
| 13 FLOW SOLUTION DAMPING CONSTNT 0.5000000 | 8 WEIGHT                   (LBM)    0.00 | 18 ** OPEN INPUT LOCATION **       0.000 |
| 11 SPLIT FRACTION OF OUTLT LEG 1 1.0000000 | 9 VOLUME                  (FT^3)    0.00 | 19 ** OPEN INPUT LOCATION **       0.000 |
| 12 SPLIT FRACTION OF OUTLT LEG 2 0.0000000 |                                        | 20 ** OPEN INPUT LOCATION **       0.000 |
| 14 HYDRAUL K FACT OF OUTLT LEG 1  0.00E+00 |                                        | 21 ** OPEN INPUT LOCATION **       0.000 |
| 15 HYDRAUL K FACT OF OUTLT LEG 2  0.00E+00 |                                        |                                          |
| ** BENCHMARK DATA                          |                                | ** OUTPUT DATA                           |
|                          MIN         NOM         MAX                |        26 OPERATION MODE FLAG           1.000                |
| INLET FLOW    (LBM/HR)     0.000     97.560     360.000              |        23 INLET FLOW     (LBM/HR)       0.000                |
| SPLIT PRESSURE  (PSIA)    14.700     14.717      14.767              |        24 SPLIT PRESSURE   (PSIA)      14.700                |
| FLOW TEMPERATURE   (F)   144.030    144.780     144.890              |        25 FLOW TEMPERATURE    (F)     144.890                |
|____________________________________________________________________|____________________________________________________________|
```

Figure A-26. Sample problem 2: SPLIT edit screen (S4).

```
CASE NAME: WCM                 SOURCE EDIT SCREEN          LAST UPDATED: 890626
SUBSYSTEM: WCM                COMPONENT: H2O              RECORD NUMBER: 000043
_____________________________________________________________________________
| ** INPUT DATA                                        * RESOURCE TRACKING *  |
| 10 SOURCE PRESSURE              (PSIA)    0.00 | 7 POWER   (WATTS)    0.00 |
| 11 SOURCE TEMPERATURE             (F)    75.00 | 8 WEIGHT    (LBM)    0.00 |
| 12 SOURCE FLOWRATE            (LBM/HR)    0.00 | 9 VOLUME   (FT^3)    0.00 |
|                                                |                           |
| CONSTITUENT EMPERICAL DATA        (^L TO OPEN)_|                           |
|_____________________________________________________________________________|

 ---------------------------------------------------------------------------
 |  LABEL   |     MASS     |  LABEL   |     MASS     |  LABEL   |     MASS     |
 |          |   FRACTION   |          |   FRACTION   |          |   FRACTION   |
 ---------------------------------------------------------------------------
 | CONST  2 |   0.0000000  | CONST 19 |   0.0000000  | CONST 36 |   0.0000000  |
 | CONST  3 |   0.0000000  | CONST 20 |   0.0000000  | CONST 37 |   0.0000000  |
 | CONST  4 |   0.0000000  | CONST 21 |   0.0000000  | CONST 38 |   0.0000000  |
 | CONST  5 |   0.0000000  | CONST 22 |   0.0000000  | CONST 39 |   0.0000000  |
 | CONST  6 |   0.0000000  | CONST 23 |   0.0000000  | CONST 40 |   0.0000000  |
 | CONST  7 |   0.0000000  | CONST 24 |   0.0000000  | CONST 41 |   0.0000000  |
 | CONST  8 |   1.0000000  | CONST 25 |   0.0000000  | CONST 42 |   0.0000000  |
 | CONST  9 |   0.0000000  | CONST 26 |   0.0000000  | CONST 43 |   0.0000000  |
 | CONST 10 |   0.0000000  | CONST 27 |   0.0000000  | CONST 44 |   0.0000000  |
 | CONST 11 |   0.0000000  | CONST 28 |   0.0000000  | CONST 45 |   0.0000000  |
 | CONST 12 |   0.0000000  | CONST 29 |   0.0000000  | CONST 46 |   0.0000000  |
 | CONST 13 |   0.0000000  | CONST 30 |   0.0000000  | CONST 47 |   0.0000000  |
 | CONST 14 |   0.0000000  | CONST 31 |   0.0000000  | CONST 48 |   0.0000000  |
 | CONST 15 |   0.0000000  | CONST 32 |   0.0000000  | CONST 49 |   0.0000000  |
 | CONST 16 |   0.0000000  | CONST 33 |   0.0000000  | CONST 50 |   0.0000000  |
 | CONST 17 |   0.0000000  | CONST 34 |   0.0000000  |          |              |
 | CONST 18 |   0.0000000  | CONST 35 |   0.0000000  |          |              |
 ---------------------------------------------------------------------------
```

Figure A.-27. Sample problem 2: SOURCE edit screen (H2O).

```
CASE NAME: WCM                                   SUM EDIT SCREEN                                  LAST UPDATED: 890621
SUBSYSTEM: WCM          COMPONENT: MIX                                                          RECORD NUMBER: 000039
_____________________________________________________________________________________________________________________
| ** INPUT DATA                                                                                                       |
| 10 FLOW DAMPING FACTOR          0.500  | 20 INPUT CONSTANT #11        0.000E+00 | 30 INPUT CONSTANT #21        0.000E+00 |
| 11 INPUT CONSTANT  #2           0.000E+00 | 21 INPUT CONSTANT #12        0.000E+00 | 31 INPUT CONSTANT #22        0.000E+00 |
| 12 INPUT CONSTANT  #3           0.000E+00 | 22 INPUT CONSTANT #13        0.000E+00 | 32 INPUT CONSTANT #23        0.000E+00 |
| 13 INPUT CONSTANT  #4           0.000E+00 | 23 INPUT CONSTANT #14        0.000E+00 | 33 INPUT CONSTANT #24        0.000E+00 |
| 14 INPUT CONSTANT  #5           0.000E+00 | 24 INPUT CONSTANT #15        0.000E+00 | 34 INPUT CONSTANT #25        0.000E+00 |
| 15 INPUT CONSTANT  #6           0.000E+00 | 25 INPUT CONSTANT #16        0.000E+00 | 35 INPUT CONSTANT #26        0.000E+00 |
| 16 INPUT CONSTANT  #7           0.000E+00 | 26 INPUT CONSTANT #17        0.000E+00 | 36 INPUT CONSTANT #27        0.000E+00 |
| 17 INPUT CONSTANT  #8           0.000E+00 | 27 INPUT CONSTANT #18        0.000E+00 | 37 INPUT CONSTANT #28        0.000E+00 |
| 18 INPUT CONSTANT  #9           0.000E+00 | 28 INPUT CONSTANT #19        0.000E+00 | 38 INPUT CONSTANT #29        0.000E+00 |
| 19 INPUT CONSTANT #10           0.000E+00 | 29 INPUT CONSTANT #20        0.000E+00 | 39 INPUT CONSTANT #30        0.000E+00 |
|                                           |                                        |                                        |
|  7 POWER                 (WATTS)    0.00 |  8 WEIGHT                (LBM)    0.00 |  9 VOLUME                (FT^3)    0.00 |
|_____________________________________________________________________________________________________________________
| ** BENCHMARK DATA                                          | ** OUTPUT DATA                                                |
|                                                            |                                                               |
| BENCHMARK   #1   0.00E+00   #2   0.00E+00   #3   0.00E+00 | 40 OUTPUT CONSTANT #1        0.00   43 OUTPUT CONSTANT #4      0.00 |
| BENCHMARK   #4   0.00E+00   #5   0.00E+00   #6   0.00E+00 | 41 OUTPUT CONSTANT #2        0.00   44 OUTPUT CONSTANT #5      0.00 |
| BENCHMARK   #7   0.00E+00   #8   0.00E+00   #9   0.00E+00 | 42 OUTPUT CONSTANT #3        0.00   45 OUTPUT CONSTANT #6      0.00 |
|_____________________________________________________________________________________________________________________
```

Figure A-28. Sample problem 2: SUM edit screen (MIX) .

```
CASE NAME: WCM                HEATER EDIT SCREEN          LAST UPDATED: 890622
SUBSYSTEM: WCM                COMPONENT: WTRHTR          RECORD NUMBER: 000009
_____________________________________________________________________________
| ** INPUT DATA                                                             |
| 10 DESIRED FLUID OUTLET TEMP (F)  145.00 |       * RESOURCE TRACKING *    |
|                                          | 7 POWER           (WATTS)    0.00 |
|        * PRESSURE DROP PARAMETERS *      | 8 WEIGHT            (LBM)    0.00 |
| 11 EQUIVALENT LENGTH        (FT)    1.00 | 9 VOLUME           (FT^3)    0.00 |
| 12 EQUIVALENT DIAMETER      (IN)    0.37 |                                 |
|_____________________________________________|_________________________________|
     | ** BENCHMARK DATA                                  | ** OUTPUT DATA
|                    MIN         NOM         MAX        |
| INLET TEMP     (F)  0.10E+03   0.10E+03   0.10E+03|13 PWR (BTU/HR)   1066.92 |
| INLET FLOW (LB/HR)   25.        25.         25.    |14 OPER TIME (HR)    1.00 |
| HEAT REQD (BTU/HR)  0.11E+04   0.11E+04   0.11E+04|                          |
|_____________________________________________________|__________________________|
```

Figure A-29. Sample problem 2: HEATER edit screen (WTRHTR).

```
CASE NAME: WCM                                   STORE EDIT SCREEN                                   LAST UPDATED: 890921
SUBSYSTEM: WCM          COMPONENT: WTRTNK                                                          RECORD NUMBER: 000474
 ** INPUT DATA
10 MAX STORAGE VOLUME       (FT^3)    2.00 |69 TEMP SET POINT      (F)    0.00 |          * THERMAL INTERFACE DATA *
11 STORAGE PRESSURE         (PSIA)   15.00 |70 TOTL LIQ/GAS VOL (FT^3)    0.00 |12 CONVECTION CONDUCTANCE   (BTU/HR/F)      0.607
17 TANK STRUCTURE MASS       (LBM)    0.30 |         * RESOURCE DATA *         |13 RADIATION CONDUCTANCE  (BTU/HR/R^4)  0.234E-09
                                           |9 VOLUME            (FT^3)    0.00 |14 MOUNTING CONDUCTANCE     (BTU/HR/F)      2.500
18 INITIAL CONTENTS MASS TEMP  (F)  145.00 |8 WEIGHT             (LBM)    0.00 |15 WALL CONDUCTANCE         (BTU/HR/F)      0.256
19 INITIAL CONTENTS MASS (LBM)      119.00 |7 POWER            (WATTS)    0.00 |16 FLUID TO WALL UA         (BTU/HR/F)     20.000
 EMPIRICAL CONSTITIUENT DATA (^L TO OPEN)_    (ITEMS 20 TO 68)
 ** BENCHMARK DATA                                  ** OUTPUT DATA
                        MIN        NOM        MAX
 CONTENTS MASS (LBM)    9.6        27.       0.12E+03 73 FINAL CONTENTS MASS (LBM)   11.96 78 TOTAL ACCUM GAS ADDED (LBM)    0.00
 CONTENTS VOL (FT^3)   0.16       0.44        1.9     74 FINAL CONTENTS VOL (FT^3)    0.20 79 TOTL ACCUM GAS VENTED (LBM)    0.00
 CONTENTS TEMP   (F)   0.14E+03   0.14E+03   0.15E+03 75 FINAL CONTENTS TEMP   (F)  144.59 80 DELTA ACCUM GAS/STEP  (LBM)    0.00
                                                      76 VENTED EXCESS MASS  (LBM)    0.00 81 ACCUM GAS SIDE MASS   (LBM)    0.00
                                                      77 FINAL TANK TEMP       (F)  143.78
```

LABEL	INT MASS FRACTION	LABEL	INT MASS FRACTION	LABEL	INT MASS FRACTION
CONST 2	0.0000000	CONST 19	0.0000000	CONST 36	0.0000000
CONST 3	0.0000000	CONST 20	0.0000000	CONST 37	0.0000000
CONST 4	0.0000000	CONST 21	0.0000000	CONST 38	0.0000000
CONST 5	0.0000000	CONST 22	0.0000000	CONST 39	0.0000000
CONST 6	0.0000000	CONST 23	0.0000000	CONST 40	0.0000000
CONST 7	0.0000000	CONST 24	0.0000000	CONST 41	0.0000000
CONST 8	1.0000000	CONST 25	0.0000000	CONST 42	0.0000000
CONST 9	0.0000000	CONST 26	0.0000000	CONST 43	0.0000000
CONST 10	0.0000000	CONST 27	0.0000000	CONST 44	0.0000000
CONST 11	0.0000000	CONST 28	0.0000000	CONST 45	0.0000000
CONST 12	0.0000000	CONST 29	0.0000000	CONST 46	0.0000000
CONST 13	0.0000000	CONST 30	0.0000000	CONST 47	0.0000000
CONST 14	0.0000000	CONST 31	0.0000000	CONST 48	0.0000000
CONST 15	0.0000000	CONST 32	0.0000000	CONST 49	0.0000000
CONST 16	0.0000000	CONST 33	0.0000000	CONST 50	0.0000000
CONST 17	0.0000000	CONST 34	0.0000000		
CONST 18	0.0000000	CONST 35	0.0000000		

Figure A-30. Sample problem 2: STORE edit screen (WTRTNK).

```
CASE NAME: WCM                                   PUMP EDIT SCREEN                                    LAST UPDATED: 890921
SUBSYSTEM: WCM          COMPONENT: WPMP                                                            RECORD NUMBER: 000285
          * MISC INPUT DATA *                  * THERMAL CHARACTERISTICS DATA *      * CENTRIFUGAL PUMP OPTION CONSTANTS *
10 MASS FLOWRATE          (LBM/HR)    10.000 | 22 SHEL-ENV CONVCT (BTU/H/F)     0.200 | 35 FLOW DAMPING FACT (0<X<1)   0.10000000
11 INLET PRESSURE           (PSIA)     0.000 | 23 SHEL-ENV RADIAT FAE (FT2)     0.045 | 13 CHARCTRISTIC CURVE CON A0  0.00000E+00
12 OUTLET PRESSURE          (PSIA)     0.000 | 24 SHEL-ENV CONDCT (BTU/H/F)     0.200 | 14 CHARCTRISTIC CURVE CON A1  0.00000E+00
18 IMPELLER EFFICIENCY    (0<X<1)      1.000 | 25 MASS-SHEL COND  (BTU/H/F)     0.500 | 15 CHARCTRISTIC CURVE CON A2  0.00000E+00
19 MOTOR EFFICIENCY       (0<X<1)      1.000 | 26 WALL-MASS COND  (BTU/H/F)    50.000 | 16 CHARCTRISTIC CURVE CON A3  0.00000E+00
20 SPECIFIED MOTR QFLAG (0=N,1=Y)      0.000 | 27 FLUID-WALL EFFECT (0<X<1)     0.900 | 17 CHARCTRISTIC CURVE CON A4  0.00000E+00
21 SPECIFIED MOTOR QLOAD (BTU/HR)      0.000 | 28 THERMAL CAPACTNCE (BTU/F)     1.000 | 36 ** OPEN INPUT LOCATION **       10.000
34 FLUID CODE(0=?,2-50=CONSTIT #)      0.000 | 29 INITIAL MASS TEMP     (F)   145.000 | 7  POWER                (WATTS)      0.000
32 ** OPEN INPUT LOCATION **           0.000 | 30 THERMAL RELAX CRITERIA    0.0000100 | 8  WEIGHT                 (LBM)      0.000
33 ** OPEN INPUT LOCATION **           0.000 | 31 THERMAL SOLUTN MAX COUNT    100.000 | 9  VOLUME                (FT^3)      0.000
                   * BENCHMARK DATA *                                                     * OUTPUT DATA *
                          MIN        NOM        MAX                                      37  MAX CAPACITY FLOW RATE       (LBM/HR)    10.000
  PUMP INLET PRESS  (PSIA)    15.0000    15.0000    15.0000                              38  PRESENT PRESSURE RISE         (PSID)     0.000
  PUMP OUTLET PRESS (PSIA)    15.0004    15.0316    15.3028                              39  PRESENT FLOW CONVERGENCE FLAG              0.000
  PUMP PRESS RISE   (PSIA)     0.0004     0.0317     0.3028                              40  PRESENT MASS TEMPERATURE          (F)   142.909
  PUMP FLOW RATE  (LBM/HR)    10.0000   132.0394   730.0000                              41  PRESENT SHELL TEMPERATURE         (F)   110.663
  PUMP POWER       (WATTS)     0.0000     0.0126     0.1473                              42  ** OPEN OUTPUT LOCATION **                0.000
```

Figure A-31. Sample problem 2: PUMP edit screen (WPMP).

```
                    CASE NAME: WCM  CONTROL EDIT SCREEN   LAST UPDATED: 941103
                                                          RECORD NUMBER: 000126
                                                          TIME UNIT FLAG (1=Hr,2=Min,3=Sec)1|PRESS
                                          SOLTN,0=Fdbk 1=Mtrx 1|
                    START TIME,   Hrs    0.000| INITIAL PRESSURES,Psia  14.700|
          | SIMULATION STOP TIME,    Hrs     1.000 | FLOW DMP FAC:M=G*dP,0<x<1 0.000E+00 |
          | SIMULATION TIME STEP,    Hrs   0.00200 | SPECIFIED PROPS CONSTIT #       0.  |
          | SIMULATION OUTPUT INTRVL,Hrs   4.00000 | SPECIFIED CP,    Btu/Lb/F 0.000E+00 |
          | CONVERGENCE CRITERION, 0<x<1 0.100E-04 | SPECIFIED DENSITY, Lb/Ft3 0.000E+00 |
          | MAX SOLUTION ITERATIONS             30 | SPECIFIED VISC,   Lb/Ft/H 0.000E+00 |
          | NUMBER OF CONSTITUENTS, x<50         9 | SPECIFIED CV,    Btu/Lb/F 0.000E+00 |
```

Figure A-32. Sample problem 2: CONTROL edit screen .

```
CASE NAME: WCM                          USER DEFINED CONSTANTS EDIT SCREEN                          LAST UPDATED: 891012
                                                                                                   RECORD NUMBER: 000003
_____________________________________________________________________________________________________________________________
|  1 DURATION (HRS) HANDWASH    0.5560E-02 | 21 FLW RATE(LB/HR) HANDWASH    180.0      | 41 ......................   0.0000E+00 |
|  2    "       "   SHOWER      0.4440E-01 | 22    "       "     SHOWER      180.0      | 42 ......................   0.0000E+00 |
|  3    "       "   DISHWASH    0.2667     | 23    "       "     DISHWASH    180.0      | 43 ......................   0.0000E+00 |
|  4    "       "   LAUNDRY     0.2778     | 24    "       "     LAUNDRY     180.0      | 44 ......................   0.0000E+00 |
|  5 PER 24 HOURS:  HANDWASH     32.00     | 25 NUMBER IN CREW              8.000      | 45 ......................   0.0000E+00 |
|  6    "       "   SHOWER       8.000    | 26 RESUPPLY FLOW  (LB/HR)      15.00      | 46 ......................   0.0000E+00 |
|  7    "       "   DISHWASH     1.000    | 27 ......................   0.0000E+00   | 47 ......................   0.0000E+00 |
|  8    "       "   LAUNDRY      2.000    | 28 ......................   0.0000E+00   | 48 ......................   0.0000E+00 |
|  9 INTERVAL (HRS) HANDWASH    0.8330E-01 | 29 ......................   0.0000E+00   | 49 ......................   0.0000E+00 |
| 10    "       "   SHOWER      0.7500     | 30 ......................   0.0000E+00   | 50 ......................   0.0000E+00 |
| 11    "       "   DISHWASH     24.00     | 31 ......................   0.0000E+00   | 51 ......................   0.0000E+00 |
| 12    "       "   LAUNDRY      8.000    | 32 ......................   0.0000E+00   | 52 ......................   0.0000E+00 |
| 13 0=ON, 1=OFF    HANDWASH.    1.000    | 33 ......................   0.0000E+00   | 53 ......................   0.0000E+00 |
| 14    "       "   SHOWER       1.000    | 34 ......................   0.0000E+00   | 54 ......................   0.0000E+00 |
| 15    "       "   DISHWASH     1.000    | 35 ......................   0.0000E+00   | 55 ......................   0.0000E+00 |
| 16    "       "   LAUNDRY      1.000    | 36 ......................   0.0000E+00   | 56 ......................   0.0000E+00 |
| 17 STRT TIME(HRS) HANDWASH    0.0000E+00 | 37 ......................   0.0000E+00   | 57 ......................   0.0000E+00 |
| 18    "       "   SHOWER      0.0000E+00 | 38 ......................   0.0000E+00   | 58 ......................   0.0000E+00 |
| 19    "       "   DISHWASH    0.0000E+00 | 39 ......................   0.0000E+00   | 59 ......................   0.0000E+00 |
| 20    "       "   LAUNDRY     0.0000E+00 | 40 ......................   0.0000E+00   | 60 ......................   0.0000E+00 |
|___________________________________________|____________________________________|___________________________________|
```

Figure A-33. Sample problem 2: USERCON edit screen.

```
CASE NAME: WCM                    PLOT EDIT SCREEN          LAST UPDATED: 890912
                                                           RECORD NUMBER: 000008
|--------PLOTSET NAME--------OPS #----------ACTIVE------^L TO OPEN-------------|
|       1.  WCMPLOTS          4            1            00081                 |
|        2.                   0            0            00000                 |
|        3.                   0            0            00000                 |
|        4.                   0            0            00000                 |
|        5.                   0            0            00000                 |
|        6.                   0            0            00000                 |
|        7.                   0            0            00000                 |
|        8.                   0            0            00000                 |
|        9.                   0            0            00000                 |
|       10.                   0            0            00000                 |
|       11.                   0            0            00000                 |
|       12.                   0            0            00000                 |
|       13.                   0            0            00000                 |
|       14.                   0            0            00000                 |
|       15.                   0            0            00000                 |
|       16.                   0            0            00000                 |
|       17.                   0            0            00000                 |
|       18.                   0            0            00000                 |
|       19.                   0            0            00000                 |
|       20.                   0            0            00000                 |
|_____________________________________________________________________________|
```

Figure A-34. Sample problem 2: PLOT edit screen.

```
                                        PLOTSET DEFINITION

CODE   COMPONENT NAME  LOCATION  STREAM#                  TITLE
----   --------------  --------  -------     --------------------------------
TIME                    0000       0        TIME, HRS
C         WPMP          0001       2        PUMP FLO RATE, LBM/HR
PRO       WPMP          0002       1        PUMP INLET TEMP, F
PRO       PR1           0002       2        HAND WASH WATER TEMP, F
PRO       PR2           0002       2        SHOWER WATER TEMP, F
PRO       PR3           0002       2        RACK PIPE 3 OUTLT TMP, F
PRO       PR4           0002       2        DISHWASH WATER TEMP, F
PRO       PR5           0002       2        LAUNDRY WATER TEMP, F
C         PR1           0001       1        HNDWSH WTR FLO, LBM/HR
C         PR2           0001       1        SHWR WTR FLO, LBM/HR
C         PR3           0001       1        RACK PIPE 3 WTR FLO, LBM/HR
C         PR4           0001       1        DSHWSH WTR FLO, LBM/HR
C         PR5           0001       1        LDRY WTR FLO, LBM/HR
PRO       S2            0002       2        SPLIT 2/STRM 2 TMP OUT, F
C         S2            0001       2        SPLIT 2/STRM 2 FLO, LBM/HR
C         S2            0001       3        SPLIT 2/STRM 3 FLO, LBM/HR
C         S3            0001       2        SPLIT 3/STRM 2 FLO, LBM/HR
C         S3            0001       3        SPLIT 3/STRM 3 FLO, LBM/HR
CON       WTRTNK        0073       0        TANK CONTENTS MASS, LBM
CON       WTRTNK        0074       0        TANK CONTENTS VOL, FT^3
CON       WTRTNK        0075       0        TANK CONTENTS TEMP, F
CON       H2O           0012       0        SOURCE FLO RATE, LBM/HR
PRO       MIX           0002       2        SUM OUTLET TEMP, F
C         MIX           0001       2        SUM OUTLT FLO RTE, LBM/HR
PRO       WTRHTR        0002       2        HEATER OUTLET TEMP, F
CON       WTRHTR        0013       0        HEATER POWER, BTU/HR
PRO       WPMP          0002       2        PUMP OUTLET TEMP, F
```

Figure A-35. Sample problem 2: PLOTSET definition screen.

A.3.3 OPS Logic Development

After the case has been constructed, the OPS logic must be created so that the components operate following the guidelines listed in section A.3.1. The basic structure of the OPS code is provided in the OPS.FOR file located in the directory containing the CASE/A code, usually [CASEA.CODE]. This file contains dummy subroutines OPS0 through OPS7. The OPS7 subroutine is used for the development of a

black box component as described in section 7.1. The remaining subroutines are called at various times during the solution routine as outlined below:

OPS0 - PRE-INITIALIZATION. This subroutine is called before the component initialization process begins. It is useful for performing specialized initializations based on user specified conditions.

OPS1 - PRE-SIMULATION. This subroutine is called at the beginning of the simulation and is used for data initialization. It is called only once during a simulation.

OPS2 - BEFORE EACH TIME STEP. This subroutine is called at the beginning of each time step and is frequently used for control of performance parameters that vary with time.

OPS3 - EACH SYSTEM ITERATION. This subroutine is called at each system iteration. It is useful for performing conditional operations such as feedback controllers.

OPS4 - END OF EACH TIME STEP. This subroutine is called just after system convergence has been obtained but before the current time has been incremented for the next step. This block is useful for performing integration functions and updating output data files.

OPS5 - EACH OUTPUT INTERVAL. This subroutine is called at each output interval as defined in the CONTROL data base. It is useful for coordinating user operations with the system output time interval.

OPS6 - END OF SIMULATION. This subroutine is called after the final time step has been completed. It is useful for post simulation summaries.

The dummy OPS file should be copied into the user's working directory for modification and is typically renamed CASENAME.FOR (i.e., for this example it is named WCM.FOR). It is suggested that all the dummy subroutines be kept together in one file even if some are not used. The modified file, as shown below, is included in VERSION 5.0 of the CASE/A program and may be found in the [CASEA.CASEA_V5.USER] directory. This program listing is heavily commented and should provide adequate explanations of all coding presented:

```
      SUBROUTINE OPS1
      INCLUDE 'CASEA$CODE:COMPCOM.INC'
C
C  OPS1 LOGIC - PRE SIMULATION. The OPS1 subroutine for this case
C  is used for initialization of data parameters and resequencing
C  of the solution procedure.
C
      COMMON /ONE/ THW,TSHWR,TDWSH,TLDRY
      COMMON /TWO/ NHW,NSHWR,NDWSH,NLDRY
      COMMON /THREE/ DT1,DT2,DT3,DT4
      COMMON /FOUR/ IFLG,JFLG,KFLG,LFLG
      COMMON /FIVE/ TSTRT1,TSTRT2,TSTRT3,TSTRT4
      COMMON /SIX/ TSTOP1,TSTOP2,TSTOP3,TSTOP4
      COMMON /SEVEN/ NCREW,TFLG1,II,ICTRH,ICTRS,ICTRD,ICTRL
      COMMON /TEST/ SRCFLO,KFLAG
      COMMON /FLORATS/ F1SET,F2SET,F3SET,F4SET
C
C  Description of input data:
C
C  THW,TSHWR,TDWSH,TLDRY are the durations, in hours, of each
C  handwash, shower, dishwash, and laundry. For example, each handwash
C  will draw hot water for 0.00556 hours for each crew member.
C
C  NHW,NSHWR,NDWSH,NLDRY specify the number of handwashes, showers,
```

```
C  etc. during a 24 hour period.
C
C  DT1,DT2,DT3,DT4 specify the time interval, in hours, between handwash,
C  shower, dishwash, and laundry cycles. Each handwash cycle consists
C  an individual handwash, lasting THW hours, for each crew member. The
C  crew members repeat the cycle every DT1 hours.
C
C  IFLG,JFLG,KFLG,LFLG are flags which are used in the OPS2 subroutine
C  to define the current state at a use point. 0 signifies that hot water
C  is being drawn at the use point. When the water supply at a use
C  point is to be turned off, the flag is set to 1.
C
C  TSTRT1,TSTRT2,TSTRT3,TSTRT4 are the initial start times, in hours,
C  in the simulation for the beginning of the handwash, shower, dishwash,
C  and laundry cycles, respectively.
C
C  F1SET,F2SET,F3SET,F4SET are the flow rates to be supplied at the
C  handwash, shower, dishwash, and laundry, respectively. When water
C  is required at a use point, it supplied at the flow rate defined
C  by the F#SET value. For example, when the handwash is in use, the
C  flow rate at the handwash is equal to F1SET. Units are lbm/hr.
C
C  NCREW is the number of crew members.
C
C  SRCFLO is the flow rate which is to be supplied from the SOURCE
C  component when resupplying water to the hot water tank. Units
C  are lbm/hr.
C
C  The GETU routine retrieves a value from the user-defined constants
C  array as determined by the location number specified as the first
C  argument of the GETU argument list. The value returned is stored
C  in the variable name defined by the 2nd argument.
C
      CALL GETU(1,THW)
      CALL GETU(2,TSHWR)
      CALL GETU(3,TDWSH)
      CALL GETU(4,TLDRY)
      CALL GETU(5,XNHW)
      CALL GETU(6,XNSHWR)
      CALL GETU(7,XNDWSH)
      CALL GETU(8,XNLDRY)
      CALL GETU(9,DT1)
      CALL GETU(10,DT2)
      CALL GETU(11,DT3)
      CALL GETU(12,DT4)
      CALL GETU(13,XIFLG)
      CALL GETU(14,XJFLG)
      CALL GETU(15,XKFLG)
      CALL GETU(16,XLFLG)
      CALL GETU(17,TSTRT1)
      CALL GETU(18,TSTRT2)
      CALL GETU(19,TSTRT3)
      CALL GETU(20,TSTRT4)
      CALL GETU(21,F1SET)
      CALL GETU(22,F2SET)
      CALL GETU(23,F3SET)
      CALL GETU(24,F4SET)
      CALL GETU(25,XNCREW)
      CALL GETU(26,SRCFLO)
C
```

```
C  Reassign certain real variable to integer variables
C
      NHW   = XNHW
      NSHWR = XNSHWR
      NDWSH = XNDWSH
      NLDRY = XNLDRY
      IFLG  = XIFLG
      JLFG  = XJFLG
      KFLG  = XKFLG
      LFLG  = XKFLG
      NCREW = XNCREW
C
C  The component solution order may be resequenced, if desired, to speed
C  the solution routine. This procedure is explained in section 7.2.
C  The order of solution is changed using the SEQUENCE command.
C  Unless the solution order is changed, the component solutions are
C  invoked in the order in which they were input into the graphical
C  schematic. In this example the entire case has been resequenced.
C  The resequencing process begins with the CALL SEQUENCE(‘SEQSTART’)
C  statement and end with the CALL SEQUENCE(‘SEQSTOP ‘) statement.
C  In between these statements, the call to the SEQUENCE routine
C  is made with the component name passed as the argument as shown
C  below. Solution is then carried out in the order in which the
C  components are listed in the SEQUENCE calls.
C
      CALL SEQUENCE(‘SEQSTART’)
      CALL SEQUENCE(‘WTRTNK  ‘)
      CALL SEQUENCE(‘WPMP    ‘)
      CALL SEQUENCE(‘S3      ‘)
      CALL SEQUENCE(‘MIX     ‘)
      CALL SEQUENCE(‘H2O     ‘)
      CALL SEQUENCE(‘WTRHTR  ‘)
      CALL SEQUENCE(‘PR3     ‘)
      CALL SEQUENCE(‘S2      ‘)
      CALL SEQUENCE(‘PS2     ‘)
      CALL SEQUENCE(‘S1      ‘)
      CALL SEQUENCE(‘PS1     ‘)
      CALL SEQUENCE(‘PR1     ‘)
      CALL SEQUENCE(‘HW      ‘)
      CALL SEQUENCE(‘PR2     ‘)
      CALL SEQUENCE(‘SHWR    ‘)
      CALL SEQUENCE(‘PS3     ‘)
      CALL SEQUENCE(‘S4      ‘)
      CALL SEQUENCE(‘PR4     ‘)
      CALL SEQUENCE(‘DSHWSH  ‘)
      CALL SEQUENCE(‘PR5     ‘)
      CALL SEQUENCE(‘LDRY    ‘)
      CALL SEQUENCE(‘SEQSTOP ‘)
C
C  Initialize the first shut-off times at each use point.
C
      TSTOP1 = TSTRT1 + THW
      TSTOP2 = TSTRT2 + TSHWR
      TSTOP3 = TSTRT3 + TDWSH
      TSTOP4 = TSTRT4 + TLDRY
C
C  Here the SETK routine is used to initialize the split fractions
C  of the SPLIT components. The SETK command allows the user to set the
C  value of a component data base parameter. In the following code, the
C  split fractions are set to 0.5 ( these values are in locations 11
```

```
C  and 12 in the SPLIT edit screen). The split fractions for each
C  SPLIT component must sum to 1.0.
C
      CALL SETK('S1      ',11,0.5)
      CALL SETK('S1      ',12,0.5)
      CALL SETK('S2      ',11,0.5)
      CALL SETK('S2      ',12,0.5)
      CALL SETK('S3      ',11,0.5)
      CALL SETK('S3      ',12,0.5)
      CALL SETK('S4      ',11,0.5)
      CALL SETK('S4      ',12,0.5)
C
C  Initialize the hot water tank temperature, mass, and conductances.
C
C  (** Note: use of comment statements, such as the ones shown below
C  at the end of the FORTRAN statements they describe, are legal in
C  VAX FORTRAN but may not be allowed in other systems. **)
C
C
      CALL SETK('WTRTNK  ',18,145.0)         ! initial contents temp
      CALL SETK('WTRTNK  ',19,119.0)         ! initial contents mass
      CALL SETK('WTRTNK  ',12,2.80)          ! init convctn conductnce
      CALL SETK('WTRTNK  ',13,0.108E-10)     ! init rad conductnce
      CALL SETK('WTRTNK  ',15,1.078)         ! init wall conductnce
C
C  Initialize flags and variables used to control the flow at
C  the use points.
C
      TFLG1  = TSTRT1
      II     = 0
      ICTRH  = 0
      ICTRS  = 0
      ICTRD  = 0
      ICTRL  = 0
C
      RETURN
      END
C
C
C
      SUBROUTINE OPS2
      INCLUDE 'CASEA$CODE:COMPCOM.INC'
C
C  OPS2 LOGIC - BEFORE EACH TIME STEP. The OPS2 subroutine will be
C  used to control the flow of hot water at the handwash, shower,
C  dishwash, and laundry facilities according to the timeline described
C  in section A.3.1.
C
      COMMON /ONE/ THW,TSHWR,TDWSH,TLDRY
      COMMON /TWO/ NHW,NSHWR,NDWSH,NLDRY
      COMMON /THREE/ DT1,DT2,DT3,DT4
      COMMON /FOUR/ IFLG,JFLG,KFLG,LFLG
      COMMON /FIVE/ TSTRT1,TSTRT2,TSTRT3,TSTRT4
      COMMON /SIX/ TSTOP1,TSTOP2,TSTOP3,TSTOP4
      COMMON /SEVEN/ NCREW,TFLG1,II,ICTRH,ICTRS,ICTRD,ICTRL
      COMMON /TEST/ SRCFLO,KFLAG
      COMMON /FLORATS/ F1SET,F2SET,F3SET,F4SET
C
C  Simulation begins with all facilities using water.
C
```

```
C  Control logic for the handwash facility.
C  ICTRH is used as a check to limit the number of handwash cycles
C  to 4 during a 24 hour period. Each handwash cycle consists of
C  a handwash of each crew member. II is a flag used to determine
C  whether or not all crew members have completed a handwash.
C
      IF (ICTRH .NE. 4) THEN
        IF (TIME .GE. TSTRT1 .AND. IFLG .EQ. 1) THEN
          IFLG   = 0
        ENDIF
        IF (TIME .GE. TSTOP1 .AND. IFLG .EQ. 0) THEN
          IFLG   = 1
          TSTRT1 = TIME + DT1           ! set start time for next
          TSTOP1 = TSTRT1 + THW         ! crew members handwash
          II     = II + 1               ! during the present cycle
          IF (II .EQ. NCREW) THEN
            TSTRT1 = TFLG1 + 4.0        ! begin the next handwash cycle
            TSTOP1 = TSTRT1 + THW       ! 4 hours after the beginning
            TFLG1  = TFLG1 + 4.0        ! of the previous cycle
            II     = 0
            ICTRH  = ICTRH + 1
          ENDIF
        ENDIF
      ENDIF
C
C  Control logic for the shower facility.
C  ICTRS is used to check if all crew members have showered. Each
C  shower is DT2 hours apart, with a shower for each crew member
C  in the 24 hour period.
C
      IF (ICTRS .NE. NCREW) THEN
        IF (TIME .GE. TSTRT2 .AND. JFLG .EQ. 1) THEN
          JFLG   = 0
        ENDIF
        IF (TIME .GE. TSTOP2 .AND. JFLG .EQ. 0) THEN
          JFLG   = 1
          TSTRT2 = TIME + DT2           ! start next shower at time
          TSTOP2 = TSTRT2 + TSHWR       ! DT2 hours after beginning
          ICTRS  = ICTRS + 1            ! of the previous shower
        ENDIF
      ENDIF
C
C  Control logic for dishwash facility.
C
      IF (ICTRD .NE. 1) THEN
        IF (TIME .GE. TSTRT3 .AND. KFLG .EQ. 1) THEN
          KFLG   = 0
        ENDIF
        IF (TIME .GE. TSTOP3 .AND. KFLG .EQ. 0) THEN
          KFLG   = 1
          TSTRT3 = TIME + DT3           ! start next dishwash at time
          TSTOP3 = TSTRT3 + TDWSH       ! DT3 hours after beginning
          ICTRD  = ICTRD + 1            ! of the previous dishwash
        ENDIF
      ENDIF
C
C  Control logic for laundry facility.
C
      IF (ICTRL .NE. 2) THEN
        IF (TIME .GE. TSTRT4 .AND. LFLG .EQ. 1) THEN
```

```
          LFLG   = 0
        ENDIF
        IF (TIME .GE. TSTOP4 .AND. LFLG .EQ. 0) THEN
          LFLG   = 1
          TSTRT4 = TIME + DT4           ! start next laundry at time
          TSTOP4 = TSTRT4 + TLDRY       ! DT4 hours after beginning
          ICTRL  = ICTRL + 1            ! of the previous laundry
        ENDIF
      ENDIF
C
C  The rest of the OPS2 logic is used to set the split fractions
C  of all the case SPLIT components based upon which facilities are
C  in operation at the present time.
C
      FLO1   = F1SET    ! required flow rate at the handwash facility
      FLO2   = F2SET    ! required flow rate at the shower facility
      FLO3   = F3SET    ! required flow rate at the dishwash facility
      FLO4   = F4SET    ! required flow rate at the laundry facility
C
C  KFLAG is used in the logic below to determine how many of the
C  facilities are active. KFLAG is initialized to 0 and then the
C  variables IFLG, JFLG, KFLG, and LFLG are checked to see if the
C  corresponding facility is active. If it is active, then KFLAG is
C  incremented. KFLAG = 4 means that all facilities are active. FLO12
C  and FLO34 are variables representing the water flow through the rack
C  standpipe to the handwash/shower and dishwash/laundry facilities,
C  respectively. FLOTOT is the flow rate to be supplied to the
C  active facilities by the recirculation loop pump component (WPMP
C  in the case schematic). FLOTOT is set below by determining
C  if a facility is active and incrementing the variable FLOTOT
C  by the flow rate required at that facility.
C
      KFLAG  = 0
      FLO12  = 0.0
      FLO34  = 0.0
      FLOTOT = 0.0
C
      IF (IFLG .EQ. 0) THEN
        KFLAG  = KFLAG + 1
        FLOTOT = FLOTOT + FLO1
      ENDIF
C
      IF (JFLG .EQ. 0) THEN
        KFLAG  = KFLAG + 1
        FLOTOT = FLOTOT + FLO2
      ENDIF
C
      IF (KFLG .EQ. 0) THEN
        KFLAG  = KFLAG + 1
        FLOTOT = FLOTOT + FLO3
      ENDIF
C
      IF (LFLG .EQ. 0) THEN
        KFLAG  = KFLAG + 1
        FLOTOT = FLOTOT + FLO4
      ENDIF
C
C  The GETK routine is used below to obtain the value stored in
C  location 36 of the PUMP component edit screen. The first argument
C  in the CALL statement is the component name and must be 8 characters,
```

```
C  left justified with blanks to fill remaining characters. The
C  second argument is an integer value which specifies the location
C  of the data base parameter of interest for that component. The
C  third argument is the variable in which the retrieved value is to be
C  stored. Input location 36 of the PUMP component edit screen is
C  used to store the rate of circulation flow to be maintained by the
C  PUMP component in the water heater recirculation loop during the
C  simulation. The recirculation loop includes the PUMP (WPMP),
C  SPLIT (S3), SOURCE (H2O), SUM (MIX), HEATER (WTRHTR), and STORE
C  (WTRTNK) components as shown in the subsystem schematic.
C
      CALL GETK('WPMP    ',36,RECRCFLO)
C
C  Make-up water is supplied to the recirculation loop from the SOURCE
C  component as long as long as water is being drawn from the hot
C  water tank. When tank contents mass returns to its initial value,
C  the SOURCE flow rate is shut off. The control logic for this is
C  contained at the beginning of the OPS4 subroutine. If no water
C  is being used at any of the facilities the pump is set to maintain
C  the recirculation flow only and the S3 SPLIT component is set to
C  divert all flow from the pump into the recirculation loop and shut
C  off flow to the rack stand-off lines. Component data base parameters
C  are changed by using the SETK routine. The arguments are the same
C  as described above for the GETK routine, except that the third
C  argument will be the value to which the component data base parameter
C  is changed.
C
      IF (KFLAG .GT. 0) THEN
        CALL SETK('H2O     ',12,SRCFLO)
      ELSE
        CALL SETK('WPMP    ',10,RECRCFLO)
        CALL SETK('S3      ',11,0.0)
        CALL SETK('S3      ',12,1.0)
      ENDIF
C
C  The logic below adjust the pump flow rate and split fractions of
C  all SPLIT components based upon how many and which facilities are
C  active. The flow rate of WPMP is adjusted to maintain the
C  recirculation loop flow rate and the flow required at the active
C  facilities. The split fraction of SPLIT S3 is then adjusted according
C  the required recirculation loop flow rate and active facility flow
C  requirements.
C
C  KFLAG = 1 corresponds to one facility using hot water. The logic
C  enclosed in the following IF/ENDIF block determines which facility
C  is active and adjusts the split fractions of the SPLIT components
C  and WPMP flow rate accordingly.
C
      IF (KFLAG .EQ. 1) THEN
        PMPFLO = FLOTOT + RECRCFLO
        FRAC1  = FLOTOT/PMPFLO
        FRAC2  = 1.0 - FRAC1
        CALL SETK('WPMP    ',10,PMPFLO) ! set pump flow rate
        CALL SETK('S3      ',11,FRAC1)  ! set fraction of leg 1 of
                                        ! SPLIT S3
        CALL SETK('S3      ',12,FRAC2)  ! set fraction of leg 2 of
                                        ! SPLIT S3
C
C  Below, each facility is checked to see if it is active. An *FLG
C  value of 0 corresponds to a facility being active. If the facility
```

```
C  is active, all other streams flows are set to 0.0 (because in this
C  block only one facility is active) and the variables FLO12 and
C  FLO34 are set to the appropriate values.
C
        IF (IFLG .EQ. 0) THEN   ! check handwash facility
          FLO12 = FLO1
          FLO2  = 0.0
          FLO3  = 0.0
          FLO4  = 0.0
        ENDIF
        IF (JFLG .EQ. 0) THEN   ! check shower facility
          FLO12 = FLO2
          FLO1  = 0.0
          FLO3  = 0.0
          FLO4  = 0.0
        ENDIF
        IF (KFLG .EQ. 0) THEN   ! check dishwash facility
          FLO34 = FLO3
          FLO1  = 0.0
          FLO2  = 0.0
          FLO4  = 0.0
        ENDIF
        IF (LFLG .EQ. 0) THEN   ! check laundry facility
          FLO34 = FLO4
          FLO1  = 0.0
          FLO2  = 0.0
          FLO3  = 0.0
        ENDIF
C
C  Set SPLIT component split fractions based upon which facility
C  was found to be active. First, SPLIT component S2   split fractions
C  are set. FLO34 is the flow rate out of leg 1 of S2 and FLO12 is
C  the flow rate out of leg 2 of S2.
C
        SPLT2L1 = FLO34/FLOTOT          ! split frac of leg 1 of S2
        SPLT2L2 = FLO12/FLOTOT          ! split frac of leg 2 of S2
        CALL SETK('S2      ',11,SPLT2L1)
        CALL SETK('S2      ',12,SPLT2L2)
C
C  Set the split fractions of SPLIT components S4 and S1.
C
        IF (FLO34 .NE. 0.0) THEN
          SPLT4L1 = FLO4/FLO34          ! split frac of leg 1 of S4
          SPLT4L2 = FLO3/FLO34          ! split frac of leg 2 of S4
          CALL SETK('S4      ',11,SPLT4L1)
          CALL SETK('S4      ',12,SPLT4L2)
        ENDIF
C
        IF (FLO12 .NE. 0.0) THEN
          SPLT1L1 = FLO1/FLO12          ! split frac of leg 1 of S1
          SPLT1L2 = FLO2/FLO12          ! split frac of leg 2 of S1
          CALL SETK('S1      ',11,SPLT1L1)
          CALL SETK('S1      ',12,SPLT1L2)
        ENDIF
C
      ENDIF
C
C  KFLAG = 2 corresponds to two facilities using hot water. The logic
C  enclosed in the following IF/ENDIF block determines which facilities
C  are active and adjusts the split fractions of the SPLIT components
```

```
C  and WPMP flow rate accordingly.
C
      IF (KFLAG .EQ. 2) THEN
        PMPFLO = FLOTOT+ RECRCFLO
        FRAC1  = FLOTOT/PMPFLO
        FRAC2  = 1.0 - FRAC1
        CALL SETK('WPMP    ',10,PMPFLO) ! set pump flow rate
        CALL SETK('S3      ',11,FRAC1)  ! set fraction of leg 1 of
                                        ! SPLIT S3
        CALL SETK('S3      ',12,FRAC2)  ! set fraction of leg 2 of
                                        ! SPLIT S3
C
C  First, the handwash and shower facilities are checked to see
C  if either one is inactive. These facilities are then checked to
C  see if both are inactive. If both of these facilities are inactive
C  then all of the flow is sent out of leg 1 of S2 to the
C  dishwash and laundry facilities. If either the shower or the
C  handwash facility is active, tests are conducted to determine
C  which stream downstream of leg 1 of S2 is active and which
C  stream downstream of leg 2 of S2 is active.
C
        IF (IFLG .EQ. 1 .OR. JFLG .EQ. 1) THEN
          IF (IFLG .EQ. 1 .AND. JFLG .EQ. 1) THEN
            FLO34 = FLO3 + FLO4
            FLO12 = 0.0
          ELSE
            IF (IFLG .EQ. 0) THEN
              FLO12 = FLO1
              FLO2  = 0.0
            ELSE
              FLO12 = FLO2
              FLO1  = 0.0
            ENDIF
            IF (KFLG .EQ. 0) THEN
              FLO34 = FLO3
              FLO4  = 0.0
            ELSE
              FLO34 = FLO4
              FLO3  = 0.0
            ENDIF
          ENDIF
        ELSE
          FLO12 = FLO1 + FLO2
          FLO34 = 0.0
        ENDIF
C
C  Set the split fraction of S2.
C
        SPLT2L1 = FLO34/FLOTOT          ! split frac of leg 1 of S2
        SPLT2L2 = FLO12/FLOTOT          ! split frac of leg 2 of S2
        CALL SETK('S2      ',11,SPLT2L1)
        CALL SETK('S2      ',12,SPLT2L2)
C
C  Set the split fractions of S4 and S2.
C
        IF (FLO34 .NE. 0.0) THEN
          SPLT4L1 = FLO4/FLO34          ! split frac of leg 1 of S4
          SPLT4L2 = FLO3/FLO34          ! split frac of leg 2 of S4
          CALL SETK('S4      ',11,SPLT4L1)
          CALL SETK('S4      ',12,SPLT4L2)
```

```
        ENDIF
C
        IF (FLO12 .NE. 0.0) THEN
          SPLT1L1 = FLO1/FLO12          ! split frac of leg 1 of S1
          SPLT1L2 = FLO2/FLO12          ! split frac of leg 2 of S1
          CALL SETK('S1      ',11,SPLT1L1)
          CALL SETK('S1      ',12,SPLT1L2)
        ENDIF
C
      ENDIF
C
C  KFLAG = 3 corresponds to three facilities using hot water. The logic
C  enclosed in the following IF/ENDIF block determines which facilities
C  are active and adjusts the split fractions of the SPLIT components
C  and WPMP flow rate accordingly.

      IF (KFLAG .EQ. 3) THEN
        PMPFLO = FLOTOT + RECRCFLO
        FRAC1  = FLOTOT/PMPFLO
        FRAC2  = 1.0 - FRAC1
        CALL SETK('WPMP    ',10,PMPFLO) ! set pump flow rate
        CALL SETK('S3      ',11,FRAC1)  ! set fraction of leg 1 of
                                        ! SPLIT S3
        CALL SETK('S3      ',12,FRAC2)  ! set fraction of leg 2 of
                                        ! SPLIT S3
C
C  When KFLAG is 3, only one stream is inactive. The logic below checks
C  to see which stream is inactive and sets the rack stand-off flow rates
C  accordingly.
C
        IF (IFLG .EQ. 1 .OR. JFLG .EQ. 1) THEN
          FLO34 = FLO3 + FLO4
          IF (IFLG .EQ. 0) THEN
            FLO12 = FLO1
            FLO2  = 0.0
          ENDIF
          IF (JFLG .EQ. 0) THEN
            FLO12 = FLO2
            FLO1  = 0.0
          ENDIF
        ELSE
          FLO12 = FLO1 + FLO2
          IF (KFLG .EQ. 0) THEN
            FLO34 = FLO3
            FLO4  = 0.0
          ENDIF
          IF (LFLG .EQ. 0) THEN
            FLO34 = FLO4
            FLO3  = 0.0
          ENDIF
        ENDIF
C
C  Set the split fractions of S2, S4 and S1.
C
        SPLT2L1 = FLO34/FLOTOT          ! split frac of leg 1 of S2
        SPLT2L2 = FLO12/FLOTOT          ! split frac of leg 2 of S2
        CALL SETK('S2      ',11,SPLT2L1)
        CALL SETK('S2      ',12,SPLT2L2)
C
        SPLT4L1 = FLO4/FLO34            ! split frac of leg 1 of S4
```

```
        SPLT4L2 = FLO3/FLO34            ! split frac of leg 4 of S4
        CALL SETK('S4      ',11,SPLT4L1)
        CALL SETK('S4      ',12,SPLT4L2)
C
        SPLT1L1 = FLO1/FLO12            ! split frac of leg 1 of S1
        SPLT1L2 = FLO2/FLO12            ! split frac of leg 2 of S1
        CALL SETK('S1      ',11,SPLT1L1)
        CALL SETK('S1      ',12,SPLT1L2)
      ENDIF
C
C  KFLAG = 4 corresponds to all facilities using hot water. The logic
C  enclosed in the following IF/ENDIF block adjusts the split fractions
C  of the SPLIT components and WPMP flow rate accordingly.
C
      IF (KFLAG .EQ. 4) THEN
        PMPFLO = FLOTOT + RECRCFLO
        FRAC1  = FLOTOT/PMPFLO
        FRAC2  = 1.0 - FRAC1
        CALL SETK('WPMP    ',10,PMPFLO) ! set pump flow rate
        CALL SETK('S3      ',11,FRAC1)  ! set fraction of leg 1 of
                                        ! SPLIT S3
        CALL SETK('S3      ',12,FRAC2)  ! set fraction of leg 2 of
                                        ! SPLIT S3
C
C  Set the split fractions of S2, S4 and S1.
C
        FLO34   = FLO3 + FLO4
        FLO12   = FLO1 + FLO2
        SPLT2L1 = FLO34/FLOTOT          ! split frac of leg 1 of S2
        SPLT2L2 = FLO12/FLOTOT          ! split frac of leg 2 of S2
        CALL SETK('S2      ',11,SPLT2L1)
        CALL SETK('S2      ',12,SPLT2L2)
C
        SPLT4L1 = FLO4/FLO34            ! split frac of leg 1 of S4
        SPLT4L2 = FLO3/FLO34            ! split frac of leg 2 of S4
        CALL SETK('S4      ',11,SPLT2L1)
        CALL SETK('S4      ',12,SPLT2L2)
C
        SPLT1L1 = FLO1/FLO12            ! split frac of leg 1 of S1
        SPLT1L2 = FLO2/FLO12            ! split frac of leg 2 of S1
        CALL SETK('S2      ',11,SPLT2L1)
        CALL SETK('S2      ',12,SPLT2L2)
      ENDIF
C
      RETURN
      END
C
C
C
      SUBROUTINE OPS3
C
C  OPS3 LOGIC - EACH ITERATION
C
30    RETURN
      END
C
C
C
      SUBROUTINE OPS4
      INCLUDE 'CASEA$CODE:COMPCOM.INC'
```

```
C
C  OPS4 LOGIC - AFTER EACH TIME STEP. The OPS4 subroutine will be
C  used for adjustment of certain parameters associated with the
C  STORE component (WTRTNK). Output parameters of interest for this
C  case are evaluated graphically using the Integrated Plot Utility.
C
      COMMON /TEST/ SRCFLO,KFLAG
C
C  Use the GETK routine to obtain the initial tank contents mass
C  and the contents mass at the present time step.
C
      CALL GETK('WTRTNK  ',19,WMASI)    ! initial tank contents mass
      CALL GETK('WTRTNK  ',73,WMAS)     ! current tank contents mass
C
C  The SOURCE flow rate into the recirculation loop is maintained
C  until the tank contents mass is returned to its original value.
C
      IF (WMAS .LT. WMASI) THEN
        CALL SETK('H2O     ',12,SRCFLO)
      ELSE
        CALL SETK('H2O     ',12,0.0)
      ENDIF
C
C  The STORE component is used to model a hot water tank which uses
C  a metal bellows to compress the fluid as it is drawn from the
C  tank. Therefore, the conductance values for the STORE component
C  thermal network must be updated to account for changes in fluid
C  to tank surface contact area as the water volume in the tank
C  fluctuates.
C
      CALL GETK('WTRTNK  ',74,SV)       ! current tank contents volume
      CALL GETK('WTRTNK  ',75,ST)       ! current tank contents temp
C
      TL    = SV*4.0/3.14159
      GWALL = 0.3996*TL + 0.157
      AO    = 5.3668*TL + 1.704
      GCONV = 0.2*AO
      GRAD  = 0.1714E-8*AO*0.05*0.9
C
      CALL SETK('WTRTNK  ',12,GCONV)    ! reset tank convection, radiation
      CALL SETK('WTRTNK  ',13,GRAD)     ! and wall conductor conductance
      CALL SETK('WTRTNK  ',15,GWALL)    ! values
C
      RETURN
      END
C
C
C
      SUBROUTINE OPS5
C
C  OPS5 LOGIC - EACH OUTPUT INTERVAL
C
      RETURN
      END
C
C
C
      SUBROUTINE OPS6
C
C  OPS6 LOGIC - POST SIMULATION
```

```
C
      RETURN
      END
C
C
C
```

A.3.4 Execution of Specialized OPS Routines

To incorporate this modified code into the CASE/A program, the WCM.FOR file must be compiled and then linked with the CASE/A library. The procedure is described in section 2, but is briefly presented here for convenience. First, compile the OPS logic FORTRAN file:

```
$ FOR/LIS/NOOP/DEBUG/CHECK=BOUNDS/CONT=45/D_LINES WCM
```

Next, the compiled code must be linked with the CASEA library. The linking process is accomplished using the LINKCAS.COM command file which should be located in user's working directory:

```
$ @LINKCAS WCM
```

The user will be notified that an executable image file has been created. This new executable is named WCM.EXE. It is this file that is executed (not CASEA.EXE) if the newly developed code is to be used. Therefore, the following commands are used to initiate a solution:

```
$ RUN/NODEB WCM

CASEA > LOADCASE WCM

CASEA_WCM > SOLVE
```

The solution proceeds until the final time is reached. The status messages are explained in the previous example. Output listings and plots may be obtained through the Integrated Plot Utility (IPU). See section 5.4 and Sample problem 1 for assistance.

APPENDIX B

CASE/A COMMAND SUMMARY

Appendix B provides an alphabetical summary of all the CASE/A program commands. This section is divided into two parts. The first section (B.1) describes all the CASE/A general commands such as component manipulation, component data base management, simulation control and execution, and utility commands. The second section (B.2) describes the commands specific to the Integrated Plot Utility (IPU) used to manage model output.

B.1 SUMMARY OF CASE/A GENERAL COMMANDS

ALARM

The `ALARM` command will issue approximately 5 seconds of BELL (CONTROL G) characters to the user's terminal. The command may be used following a lengthy operation to alert the user to the operation completion.

Format: `ALARM`

ARCHIVE

The `ARCHIVE` command initiates the Archive Process. This provides the capability to archive the current case data simulation solution to an ASCII file whose name is of the form:

`CASENAME_ARCHIVESETNAME.ARC - where ARCHIVESETNAME is the name specified in the Archive data base`

During the Archive Process, the data in the case "MOD" file, CONTROL, LABELS, USERCON, PLOT, and ARCHIVE data bases are copied to an ASCII file along with the OPS code (if any).

The user has the option to archive plotset data associated with this case. If selected, each plotset designated for archiving by a "1" in the ARCHIVE column of the ED PLOT screen is archived. The plot data is copied to a file of the form:

`CASENAME_ARCHIVESETNAME_PLOTSETNAME.DAT`

The archive files will be placed in a directory of the user's choice. If no directory is specified, the process will create an archive directory which is a subdirectory of the current directory.

Format: `ARCHIVE`

ASSIGN

The `ASSIGN` command assigns a subsystem to a cabin or module. This provides the capability to address multiple cabins/modules with different configurations and/or environments. The `ASSIGN` command must be issued before executing `SOLVE`. If the user fails to make a cabin or module assignment, the cabin air, structure, and mean radiant environment will default to 75°F from a component standpoint. The user should zero out the conductances for the components that are assumed to operate adiabatically. If the interaction with the environment is disabled, then the `ASSIGN` command is not required.

Format: `ASSIGN;SSNAME;CABINNAME or MODULENAME`

AUTOPLOT

The AUTOPLOT command initiates the Autoplot Process. This provides the capability of plotting all the data in a plotset and store the plots in a Tektronix 4014 format. The plots can then be printed or reviewed. The Autoplot Process can run in interactive mode or in batch mode as selected by the user. If the plotset contains many dependent variables, it is more practical to run the Autoplot Process in batch mode. Each dependent variable is plotted on a separate page in the Autoplot Process. If it is run in batch mode, a command file is generated and the job is submitted to the batch queue. In either interactive mode or batch mode, a plot file is generated for each plotset in the current case in which a "1" is inserted in the AUTOPLOT column of the ED PLOT screen. Each plot file name is of the form:

CASENAME_PLOTSETNAME.PLT

Format: AUTOPLOT

CN

The CN (connect) command allows the user to specify an interconnect path between components. The crosshairs are activated and the user specifies a starting point by positioning the crosshairs within the small "hit box" on the component stream and typing any character. The user then defines the path to the next component. If intermediate points are desired (segmented line) the user designates the intermediate point locations by typing any character at the desired location(s). The user designates the termination point by placing the crosshairs within the second component stream "hit box" and typing any character. The path is drawn on the screen as it is defined. If an existing path is redefined, the old path will disappear on the next RD (redraw) command. The crosshair movement on some terminal types can be sped up by holding down the shift key in conjunction with the desired cursor key. The two arguments LEQ and DEQ are optional connection parameters used in the calculation of the pressure drop through the connection. The parameter LEQ is the hydraulic equivalent length, in feet, of the connection.

The parameter DEQ is the hydraulic equivalent diameter, input as hundredths of an inch (i.e., DEQ = 100 for an equivalent diameter of one inch). The absence of either argument results in a pressure drop of zero for the connection.

The CN command is unique for the CNTRLLR and TIMER components. These components require the user to connect data streams to "hit dots" located either on the component icon body or on the inlet or outlet component streams. The data streams appear as dotted lines on the subsystem screen. See Sections 7.6 and 10.0 for complete discussions of the CNTRLLR and TIMER components.

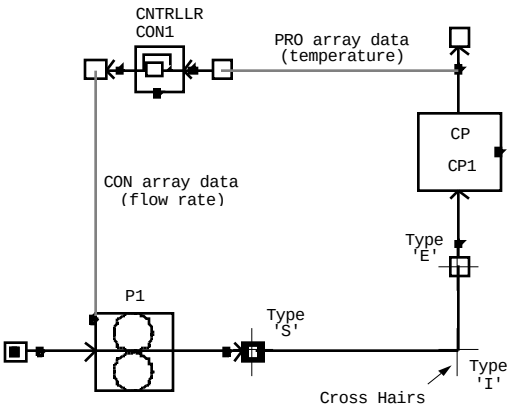
Format: CN[;LEQ;DEQ] (fluid streams)

CN (data streams)

Notes: The two arguments LEQ and DEQ for the fluid stream CN command are optional; the default pressure drop of a connection is 0.0.

Example: A pump is located on the screen and it is desired to connect a fluid stream between the pump outlet and the cold plate inlet. To do this, the CN command is entered and crosshairs appear on the screen. The crosshairs are placed on the pump outlet stream hit box and any character is typed. The same procedure is followed to locate the intermediate point and the end point as shown below.

A CNTRLLR component is located on the screen and it is desired to control the pump flow rate based on the cold plate outlet stream temperature. The CNTRLLR input data (PRO array) stream is connected between the CNTRLLR input hit box and cold plate outlet stream hit dot. Similarly, a data (pump CON array) stream is connected between the CNTRLLR outlet hit box and the pump icon body hit dot as shown below:



COPYALL

The COPYALL command will allow the user to copy a portion (or all) of a subsystem screen to a new or existing subsystem screen by drawing a box around the components to be copied. The command argument is the destination subsystem. After the command is entered, the crosshairs (cursor) will appear three times. The first appearance allows the user to designate one of the box corners by typing any character. The second appearance of the cursor allows the user to designate the other box corner (diagonally opposing the first corner) by typing any character. After the second box corner is determined, the box is drawn on the subsystem screen with a small marker in the center of the box. The third appearance of the cursor is used to designate the location of the box ‘center-marker’ on the object (destination) subsystem. The user moves the crosshairs to the new location and types the character “C” for continue. If the box is to appear in the same location on the object screen as it appears in the original subsystem screen, then the character “D” (for default) is typed. If any other character is typed at this point, the COPYALL command is terminated. Connections are copied, if both endpoints are on components inside the box. Any notes and labels within the box are also copied. The user is prompted for new equipment names for the copied components (eight characters maximum). A warning will appear on the screen if the new name is a duplicate of any other component name in the case.

Format: COPYALL;SSNAME

Notes: The maximum number of components that can be grouped and copied is 50. The BLACKBOX component cannot be copied.

D

The D (delete) command allows the user to delete a portion or all of a subsystem screen by drawing a box around the components to be deleted. After the command is entered, the crosshairs will appear two times. The first appearance allows the user to designate one of the box corners by typing any character. The second appearance of the cursor allows the user to designate a box corner diagonally opposed to the first corner and the box will be drawn around the components to be deleted. The user then types the character “D” to delete the selected components or any other character to abort the command and return to the command prompt. The deleted components will disappear on the next subsystem RD (redraw) command.

Format: D

DCL

The DCL command allows the user to enter the VAX DCL (Digital Command Language). The user may then execute any DCL level command with the exception of changing the default directory. Operations may be performed on files outside the current directory, however, by simply using the appropriate disk and directory references in the commands. The user returns to the CASE/A environment by entering LOGOUT or LO at the DCL prompt.

Format: DCL

DELCASE

The DELCASE (delete case) command will delete an existing case. All data base records associated with the case will be deleted. The user must delete the CASENAME.MOD file and any OPS logic FORTRAN file associated with the case from the DCL environment after leaving CASE/A.

Format: DELCASE

Note: This command must be invoked from the 'CASEA_CASENAME >' case prompt. The current active case will then be deleted.

DELCN

The DELCN (delete connection) command allows the user to delete a connection from the existing subsystem without affecting component placement. The crosshairs are activated and the user designates the starting location of the connection by placing the crosshairs on the component stream and typing any character. The crosshairs remain active and the ending point is designated in a similar manner.

The connection will disappear on the next RD (redraw) command.

Format: DELCN

DELEQ

The DELEQ (delete equipment) command is used to delete a component from the existing subsystem along with its connections. The crosshairs are activated and the user designates the component to be deleted by placing the crosshairs within the figure and typing any character. The component and all its connections will disappear on the next RD (redraw) command.

Format: DELEQ

DELLAB

The DELLAB (delete label) command is used to delete an existing label. The crosshairs are activated and the user designates the label to be deleted by placing the crosshairs on the label and typing any character. The label will disappear on the next RD+ (redraw plus) command.

Format: DELLAB

DELNOTE

The DELNOTE (delete note) command allows the user to delete an existing note from a subsystem screen. The crosshairs are activated and the user designates the note to be deleted by placing the crosshairs on the note and typing any character. The note will disappear on the next RD (redraw) of the subsystem.

Format: DELNOTE

DIR

The DIR (directory) command allows the user to identify all subsystems and components associated with a specific case. When this command is invoked, each subsystem name along with the components and their respective names are displayed on the screen.

Format: DIR

E
(screen editor)

The E (edit with cursor) command allows the user to edit the data base edit screen for a particular component in the system. The user must be on a composite graphics/text terminal (or emulation) with the TERM command set to "6". After the E command is entered, the crosshairs (cursor) are activated and the user designates the component to be edited by placing the cursor on that component and typing any character. The screen editing mode is activated and the arrow keys are used to advance to the desired parameter location and a new value is entered. A <Ctrl Z> saves the data values and exits from the editor. The data is stored permanently for future reference. If the user decides not to save the adjusted values, then the F11 key (or <Ctrl X>) is typed.

Format: E

Notes: Functionally equivalent to the ED command.

E
(line editor)

The E (edit with cursor) command allows the user to change data in line edit mode for a particular component in the system. This mode of editing is the default mode when entering a CASE/A model (TERM;1). After the E command is entered, the crosshairs (cursor) are activated and the user designates which component is to be edited by placing the cursor on that component and typing any character. The current component data base input parameters are displayed, 19 items per screen, and the user designates changes by an 'item number; new value' format. The user may page down in the display by entering "1". The edit session can be terminated at any time by entering a "0". The updated contents are then displayed.

Format: E

Notes: Functionally equivalent to the ED command.

ECHO and NOECHO

The ECHO command activates the command line echo which repeats the user-supplied command. The echo property is turned off by default and may be deactivated using the NOECHO command. The echo is generally turned off when running PLOTs to avoid having the next command appear on the plot. The status of the echo flag may be checked using the FLAGS command.

Format: ECHO or NOECHO

ED
(screen editor)

The ED (edit without crosshairs) command allows the user to edit the data base edit screen for a particular component in the system. The user may be on a text-only or composite terminal when issuing this command. The TERM command must be set to "2", "3", "4", or "6" depending on which type terminal is being used (see Section 6.1 for explanation of TERM command). This command has the same options and syntax of the E command but the user must indicate the type or name of the component to be edited. If more than one component type is present in a case, then CASE/A prompts the user with a numbered list of the component names of that particular equipment type. The user then enters the number of the component that is desired for editing.

Format: ED;EQTYPE (option 1)
ED;EQNAME (option 2)

Examples: ED;CHX (prompts the user with a list of all CHX components in the case if more than 1 exists).

ED;LABHX (edits directly the CHX component name LABHX).

Notes: EQTYPE is the mnemonic which is associated with the equipment type (see Table 10.0-1). EQNAME is the name given to the component by the user.

ED
(line editor)

The ED (edit without cursor) command allows the user to change data in line edit mode for a particular component in the system. The user may be on a graphics or text terminal when issuing this command. This command has the same options and syntax of the E command but the user must indicate the type of component to be edited. This command can be used in the batch mode if required. If more than one component type is present in a case, then CASE/A prompts the user with a numbered list of the component names of that particular equipment type. The user then enters the number of the component that is desired for editing.

Format: ED;EQTYPE (option 1)
ED;EQNAME (option 2)

Examples: ED;CHX (prompts the user with a list of all CHX components in the case if more than 1 exists).

ED;LABHX (edits directly the CHX component name LABHX).

Notes: EQTYPE is the mnemonic which is associated with the equipment type (see Table 10.0-1). EQNAME is the name given to the component by the user.

EDT

The `EDT` command enters the VAX/VMS text editor allowing the user to examine the output file `CASENAME.LPP` or modify the operations file `CASENAME.FOR`. Any text file in the current directory is available to the user. All of the VAX/VMS editor commands are usable. When the user types "`EXIT`" or "`QUIT`" to leave the editor, control is returned to CASE/A.

Format: `EDT[;filename.ext]`

Notes: If the argument is omitted, the `CASENAME.LPP` file is opened.

EXIT

The `EXIT` command will terminate a CASE/A session and automatically save the current model configuration to the CASENAME.MOD file. The user will be returned to the DCL "`$`" prompt.

Format: `EXIT`

FASTED
(screen editor)

The `FASTED` (fast edit) command allows the user to change, with a single command, input parameters for all components of a particular type (i.e., CP, PUMP, CHX, etc.) in a case. The edit screen of the particular component type that was located first during subsystem screen construction will appear. As changes are made to this screen, they will automatically be made for the same item number in all the other records of that component type in the case. It is important to note that *individual item numbers only* are changed throughout the case; carbon copies of the edit screen are not made in order to protect other user data.

Format: `FASTED;EQTYPE`

Example: `FASTED;CP` (allows the user to change selected items for all the CP [cold plate] components in a case).

Notes: Due to the limitations of the overstrike method of editing component edit screens, the item changed must differ from the previous value in some way to assure changes in all the other data base records. For example, if an item value is 10.0 in the edit screen that first comes up, the user may not change all other similar items to 10.0 by simply typing 10.0 over the existing number. The editor will not recognize the change. The user may, however, change the item to 10.1 and then back to 10.0 to assure changes are made across the board.

FASTED
(line editor)

The `FASTED` (fast edit) command allows the user to change, with a single command, input parameters for all components of a particular type (i.e., CP, PUMP, CHX, etc.) in a case. The user may be on a graphics or text terminal when issuing this command. Changes are specified in a "Item #;New Value" format and are terminated by entering a "0".

Format:
```
FASTED;EQTYPE
item #;new value
item #;new value
.
.
.
0
```

FLAGS

The FLAGS command provides a listing of the current state of the operation control flags. An example of the FLAGS summary follows (these are the system default settings). The commands to set each flag "on" or "off" are provided in the last two columns.

```
 SYSTEM FLAG SUMMARY          |   on      off
                              |
 1 LOGO                  ON   |  LOGO   ,NOLOGO
 2 INVERSE FLAG          OFF  |  INVERSE,NORMAL
 3 COMMAND ECHO          ON   |  ECHO   ,NOECHO
 4 AUTO HARD COPY        ON   |  COPY   ,NOCOPY
 5 PLOT TIME             OFF  |  TIME   ,NOTIME
 6 DUAL Y AXES           OFF  |  DUAL   ,NODUAL
 7 VECTOR GRAPHICS       OFF  |  VECT   ,NOVECT
 8 X AXIS LOG SCALE      OFF  |  LOGX   ,NOLOGX
 9 Y AXIS LOG SCALE      OFF  |  LOGY   ,NOLOGY
10 OVER-WRITE PROTECT    OFF  |  REUSE  ,NOREUSE
11 BOLD PLOT             OFF  |  BOLD   ,NOBOLD
12 PAUSE FOR COPY        OFF  |  PAUSE  ,NOPAUSE
13 NASA LOGO             OFF  |  NASA   ,NONASA
14 COLOR FLAG            OFF  |  COLOR  ,NOCOLOR
13 TASK COMPLETION EST.  OFF  |
13 PLOT LEGEND           ON   |  LEGEND ,NOLEGEND
```

Format: FLAGS

HELP

The HELP command will provide a listing of all available CASE/A commands. The listing is divided into three distinct categories: 1) Model Development and Solution Commands, 2) Data Management Commands, and 3) Engineering Analysis Commands. To display information regarding a specific command, simply execute HELP;command. At the conclusion of any help session, the user is automatically returned to the command prompt.

Format: HELP[;command]

LAB

The LAB (label) command allows the user to display stream or component characteristics on the subsystem schematic. This command is normally used after a system solution. To initiate this command, three arguments are required, as shown in the format below. NC represents the desired constituent, property, or data base parameter as defined in Table 7.2-1 for the Constituent array, Table 7.2-2 for the PROperty array or the solution routine overviews (Section 10) for the component data base CON array. LABEL is the title to be printed out with the value and is limited to eight characters. INA represents which array is being referenced, 1 for Constituent array, 2 for PROperty array, and 3 for CON array. Once the command has been activated, the crosshairs will appear whereupon the user can specify the desired stream and component to which the label applies by placing the crosshairs in the hit box of that stream, or on the component for the CON array values, and typing any character. The crosshairs will again appear to allow the user to specify the desired location of the label by typing any character with the crosshairs at the desired location.

Format: LAB;NC;LABEL;INA

LO

The LO (locate) command is used to locate a new component in the existing subsystem. The crosshairs are activated to allow the user to define a location for the component on the screen. To define a location, the user types any single character with the crosshairs at the desired location. The component figure is then drawn. A unique name for each component is suggested. This will simplify any logic in the OPS blocks which must refer to this component.

Format: LO;EQTYPE;EQNAME

Example: LO;CHX;CHX1 (locates a condensing heat exchanger named "CHX1")

Notes: EQTYPE is the mnemonic associated with the equipment type. A list of the legal equipment types (EQTYPE) is provided in Table 10.0-1. The equipment names (EQNAME) may be up to 8 characters, however, this may exceed the space available in some component icons.

LOADCASE

The LOADCASE command will load an existing case into memory. This command is used to load an *existing* case only. If the user is already in an existing case, this command will switch the user from the current case to the requested second case. The current case will be automatically saved before loading the new case.

Format: LOADCASE;CASENAME

Note: This command may be invoked from either the "CASEA >" system prompt or the "CASEA_CASENAME >" case prompt.

M

The M (move) command allows the user to move a portion (or all) of a subsystem screen to a new position on the subsystem screen by drawing a box around the components to be moved. Connections and intermediate points of connections inside the box will also be moved. Any notes and labels within the box are also moved. After the command is entered, the crosshairs (cursor) will appear three times. The first appearance allows the user to designate one of the box corners by typing any character. The second appearance of the cursor allows the user to designate the other box corner (diagonally opposing the first corner) by typing any character. After the second box corner is determined, the box is drawn on the subsystem screen with a small marker in the center of the box. The third appearance of the cursor allows the user to designate the location of the box "center-marker" at the new subsystem screen location by typing the character "C" for continue. If any other character is typed at this point, the M command is terminated.

Format: M

Notes: The maximum number of components that can be grouped and moved is 50.

MERGE

This command allows the user to include a complete subsystem (or the entire model) from a different case in the current case. Complete subsystems can be built and verified as separate cases and then the subsystem merged into the larger model. Development time is reduced since the models are developed as smaller, more manageable problems and then assembled into a single model. The routine will check for duplicate component names and solicits a new name from the user if necessary. If the user does not want to change the name, a carriage return is entered and execution continues.

Format: `MERGE;CASENAME;SSNAME`

NOTE: If the case name or subsystem name is in error or omitted, the program will solicit a new input. The routine prompts the user to enter the location of the source CASENAME.MOD file and the source data base files. These locations must be of the following form:

```
Host "username password"::
Device:[Directory]
```

Host and device are necessary only if different from current case.

MV

The `MV` (move) command allows the user to move an existing component to relieve congestion or otherwise improve the subsystem layout. The crosshairs are actuated and the user designates a component by placing the crosshairs within the figure and typing any character. The crosshairs remain active and the user designates a new location by again typing any character. The figure will appear in the new location and the connections are dragged with it upon the next subsystem `RD` (redraw) command.

Format: `MV`

MVLAB

The `MVLAB` (move label) command allows the user to move an existing label to relieve congestion or otherwise improve the subsystem appearance. The crosshairs are actuated and the user designates a label by placing the crosshairs on the label and typing any character. The user designates a new location for the label by moving the crosshairs and typing any character. The label will appear in the new location immediately and the old label will disappear after a RD command.

Format: `MVLAB`

MVNOTE

The `MVNOTE` (move note) command allows the user to move an existing note to relieve layout congestion or otherwise improve the appearance of the subsystem screen. The crosshairs are actuated and the user designates a note by placing the crosshairs on the note and typing any character. The crosshairs remain active and the user designates a new location for the note by again typing any character. The note will appear in the new location immediately and the old note will disappear on the next RD command.

Format: `MVNOTE`

NEWCASE

The NEWCASE command will initialize a new case. The user may then begin the process of naming subsystems and locating components and connections. If the user is already in an existing case, this command will switch the user from the current case to the new case. The current case will be automatically saved before entering the new case.

Format: NEWCASE;CASENAME

Note: This command may be invoked from either the "CASEA >" system prompt or the 'CASEA_CASENAME >' case prompt.

NOTE

The NOTE command allows the user to place a note on a subsystem screen. The note appears immediately after completion of the NOTE command and will appear with the subsystem whenever the subsystem is displayed. After the command has been initiated, the screen crosshairs (cursor) are activated and the user can place the note in the desired location by typing any character. The required command arguments are shown in the format definition below. NT determines the string output format and the allowable options for NT are as follows:

L - Each string is left-justified from the cursor location, starts as a new line, and can contain a maximum of 20 characters.

C - Each string is centered over the cursor location, starts as a new line, and can contain a maximum of 20 characters.

N - Only one string is allowed, it is left-justified from the cursor location, and can contain a maximum of 80 characters.

CSZ determines the character size of the note strings and the allowable options for CSZ are: 1, 2, 3, or 4, where 4 designates the smallest character size.

Format: NOTE;NT;CSZ;STRING1; STRING2;STRING3;STRING4

Caution: The semicolon delimiter should always be used with this command to preserve blank spaces in the note itself.

Example: NOTE;L;4;This is note 1;This is note 2;This is note 3;

PREPRO

The PREPRO command creates a more efficient OPS logic file by substituting actual component addresses (calculated during the preprocessing operation) for component data retrieval and modification. See Section 7.5 for information on the OPS logic file and the data retrieval and modification routines.

Format: PREPRO;FILENAME

Notes: The preprocessor uses FORTRAN logical units 15 and 16. The FILENAME.FOR file will be written over each time preprocessing is attempted. The maximum number of characters for FILENAME is eight.

PRINTSS

The PRINTSS command allows the user to obtain a printout of the current data base parameters for every component in a subsystem. Upon execution of the PRINTSS command, the component parameters are written to a file which is submitted to the system line printer spooler. This file is automatically deleted after printing.

Format: PRINTSS;SSNAME

Notes: The PRINTSS command uses FORTRAN logical unit 18 for creation of the output file.

RETRIEVE

The RETRIEVE command initiates the retrieval of an archived case into the current case. This process copies the case data from the ASCII file located in the archive subdirectory into the case the user has currently loaded. The loaded case can be empty or one that contains data. The user is prompted at each stage of the process regarding whether he/she wishes to overwrite the data in the case "MOD" file, the CONTROL data base, the LABELS data base, the USERCON data base, the PLOT data base and the ARCHIVE data base.

Note this is the reverse operation of the ARCHIVE Command - refer to the ARCHIVE command for additional information.

Format: RETRIEVE

RD and RD+

The RD (redraw) command is used to refresh the current subsystem screen. This may be invoked at any time to clear unwanted messages from the screen and show the effect of component moves and deletions. RD+ (redraw plus) is used with the LAB command so that the labels generated are included on the screen. Component stream connection hit boxes are omitted with the use of the RD+ command.

Format: RD or RD+

RNEQ

The RNEQ (rename equipment) command allows the user to change the name of a component on a subsystem screen. The second argument of the command is the new equipment name. After the command is entered, the crosshairs are activated and the user designates the component in the subsystem to be renamed by placing the cursor on that component and typing any character. The new name will appear upon the next subsystem RD (redraw) command.

Format: RNEQ;NEWEQNAME

Notes: The new equipment name can contain a maximum of eight characters. A warning will appear on the screen if the new name is a duplicate of any other component name in the case.

RNSS

The RNSS (rename subsystem) command will allow the user to change the name of the subsystem screen that is currently activated. The command argument is the new subsystem name. The new subsystem name will appear upon the next RD (redraw) of the subsystem.

Format: RNSS;NEWSSNAME

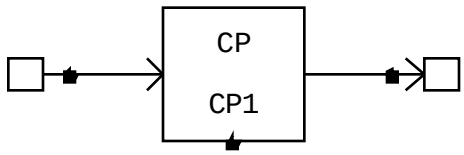
Notes: All components, notes, labels, and cabin assignments associated with the subject subsystem are updated accordingly.

ROTATE

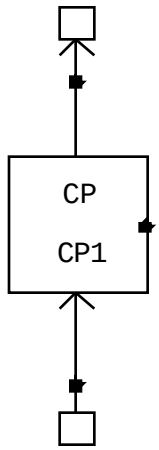
The ROTATE command will allow the user to rotate an existing component. Once the command has been activated, the crosshairs will appear and the user can specify the component to be rotated. This is accomplished by typing any single character with the crosshairs placed on the desired component. The component will be in the new orientation on the next RD (redraw) command. To initiate this command, one argument is required as shown in the format below. ANGLE represents the degree of rotation that is desired. The rotation angle must be a multiple of ±90˚ with a magnitude not greater than 360˚. A positive angle represents a counter-clockwise rotation.

Format: ROTATE;ANGLE

Example: The command LO;CP;CP1 would draw the following default cold plate component icon orientation on the screen:



A subsequent ROTATE;90 command would activate the crosshairs on the screen. The user would then place the crosshairs on the component icon to be rotated and type any keyboard character. Upon the next RD command, the icon would appear rotated 90° counterclockwise as follows:



SAVE

The SAVE command will save the current case to the CASENAME.MOD file. This is performed automatically following a component location (LO) command and during the file wrap up for the EXIT command.

Format: SAVE

SAVEAS

The SAVEAS command will duplicate an existing case in its entirety and save it under a new name. The user must be in an existing case to invoke this command. The old case will be automatically saved before entering the new case.

Format: SAVEAS;NEWCASENAME

SOLVE

The SOLVE command initiates a solution of the system. To conserve computer resources, the solution routine can be limited to only temperature (T) or pressure (P) solutions by placing an argument after the solve command. If both solutions are desired, then no argument is required (defaults to both). A time status of the solution progress is displayed on the screen along with any diagnostic messages. Control parameters for the solution are contained in a "CONTROL" file which may be edited using the ED command (i.e., ED;CONTROL). This file contains the start time, stop time, time step, output interval, iteration loops, and relaxation convergence criteria. A comprehensive listing of results at the user-specified output interval is written to a file with a name equal to the case name with a ".LPP" extension.

Format: SOLVE;X

Notes: If X is not equal to P or T then the default is both.

SS

The SS (subsystem) command allows the user to define the subsystem within the current case upon which subsequent operations are to be performed such as adding, moving, or deleting components or establishing or modifying connections. Components on different subsystem screens may be connected with the bubble component feature described in Section 10.

Format: SS;SSNAME

Notes: The SSNAME field is limited to 8 characters. An automatic redraw (RD) is executed upon issuance of the SS command.

TERM

The TERM (terminal) command defines the type of terminal used in a given CASE/A work session to permit the proper interpretation of editing and output commands. The following is a summary of the terminal types currently supported.

TERM;1 - Tektronix 4014

TERM;2 - VT100 and VT200

TERM;3 - DUAL VT100/4014

Format: TERM;i

TYPE

The TYPE command allows the user to view the component data base for a particular component in the system. The user specifies the type of component to be viewed via the command argument. If multiple components of this type exist in the subsystem, the user is asked to designate the component of interest (a program prompt is provided with a list of choices). The current contents of the component data base are displayed.

Format: TYPE;EQTYPE

UNASSIN

The command unassigns a subsystem from a cabin. It is functionally opposite of the `ASSIGN` command.

Format: `UNASSIN;SSNAME`

VECT and NOVECT

The `VECT` (vector) command enables the use of vector characters. The `NOVECT` (no vector) command disables the use of vector characters.

Format: `VECT` or `NOVECT`

B.2 SUMMARY OF INTEGRATED PLOT UTILITY (IPU) COMMANDS

AVERAGE

The AVERAGE command generates an integrated average of all items in the data base at the specified time interval. The results are written to the alternate data base number (1, 2, 3, or 4) specified. The alternate data base is opened prior to the execution of the routine. The data base should usually have the same data definition file (.DDF) contents as the original but under a different name. The active set will contain the record numbers of the averaged data set. The user must identify the item location of the time variable in the record. The results appear as stepwise data on the specified interval.

Each step has both a start time and a stop time written to the alternate data base. This results in the end time of the Nth step coinciding with the starting time of the (N+1)th step. If the time step is greater than the time step of the data in the data base, a trapezoidal rule integration is performed on the data points within the interval. Appropriate interpolation is applied to assure that the first and last points in the time step are properly handled under the condition that only part of a data interval falls within the time step in question. If the time step is smaller than the time step of the data in the data base, the averaged data will appear as a stepwise linear interpolation between the points which exist.

Format: `AVERAGE;timestep;alternate DB number;time item`

BOLD and NOBOLD

The BOLD command results in IPU plots being displayed with a bold line. This was found to result in better readability particularly when the plot is to be used for presentation view graphs. The BOLD feature is on by default. In some circumstances (extremely cyclical data) the heavy line is not desirable because the bold lines obscure needed detail in the plot. The feature may be deactivated by the NOBOLD command in these cases.

Format: `BOLD` or `NOBOLD`

CALC

The CALC (calculation) command allows the user to mathematically modify the data base contents in accordance with a user-specified algorithm. The CALC function is designed to emulate a RPN (Reverse Polish Notation) calculator key-stroke programming method (Hewlett-Packard calculator). The CALC function operates on each record in the active set. The CALC function has a 10 register stack and 10 memory cells available for use in accomplishing the calculations specified. The stack operates in the following manner:

- A retrieved item or user specified constant is placed into the lowest stack location, pushing all other data in the stack up one location.
- Two number operations (addition, multiplication, etc.) operate on the bottom two locations. Both numbers are destroyed in the process with the result of the operation in the lowest stack location. The data which was in the third location and above fall one location.
- One number operations (Sine, Log, 1/x, etc.) operate on the lowest location. The operand is destroyed with the result now occupying the lowest location.
- A command to store data takes the information from the lowest location to a place in the specified data base item or storage cell. The stack contents are not affected.

CALC (Cont.)

The user begins the process by issuing the `CALC` command. The user then provides the "key-stroke" sequence which includes the combinations of operations and arguments from the available operation set described below. Each operation is submitted on a separate line. The user terminates the input of the calculation sequence with `<Ctrl Z>`. The calculation sequence is then applied to each record in the active set. It is recommended that unusually long or complex sequences be set up to be executed in the form of a macro using the `RUN` command. The routines `CLCALC` (clear memory and "registers") and `DCALC` (display memory and "registers") can be used before and after the `CALC` execution as required. The user should treat a `CALC` sequence as any other program which operates on valuable data base information. Copies of the data base should be made prior to "debugging" the algorithm if any doubt exists as to the proper operation of the sequence to allow recovery from a malfunctioning command sequence.

The following provides an overview of the available `CALC` operations. Following the convention frequently used in the Hewlett-Packard manuals the lowest two stack register locations will be designated as: x for register 1, and y for register 2.

`I;item`	Places the specified item in x, pushes the rest up one register.
`M;location`	Places the contents of the memory location specified into x, pushes the rest up one register. The location argument may be an integer from 1 to 10.
`SI;item`	Places the contents of the x position into the data base at the item specified.
`SM;location`	Places the contents of the x position into the memory at the location specified (integer from 1 to 10).
`GOTO;step #`	Transfers control to the specified step number.
`E;value`	Places the constant "value" into x, pushes the rest of the registers up in the process
`IFEQ;step`	If register x .EQ. register y go to the specified step
`IFLT;step`	If register x .LT. register y go to the specified step
`IFLE;step`	If register x .LE. register y go to the specified step
`IFGT;step`	If register x .GT. register y go to the specified step
`IFGE;step`	If register x .GE. register y go to the specified step
`IFNE;step`	If register x .NE. register y go to the specified step
`+`	Add x and y, x and y are "destroyed", x = sum
`-`	Subtract x from y, x and y are "destroyed", x= difference
`*`	Multiply x and y, x and y are "destroyed", x = product
`/`	Divide x into y, x and y are "destroyed", x = quotient
`**`	Raise y to x power, x and y are "destroyed", x = result

CALC (Cont.)

SIN	x = SIN(x), x in radians
COS	x = COS(x), x in radians
TAN	x = TAN(x), x in radians
ASIN	x = ARCSIN(x), x in radians
ACOS	x = ARCCOS(x), x in radians
SINH	x = SINH(x)
COSH	x = COSH(x)
TANH	x = TANH(x)
EXP	x = EXP(x)
LN	x = LN(x)
SQRT	x = SQRT(x)
LOG	x = ALOG10(x)
1/X	x= 1/x
X-Y	x=y , y=x
CF	convert x to °F from °C
FC	convert x to °C from °F
ABS	X = ABS(x)
END	end program

Format: CALC

CHANGE

The CHANGE command is used to edit alphanumeric data types. The routine will replace the old string with the new string in the designated item for all records in the active set. The data field will be expanded or contracted as required to accommodate the new string length. If the expansion results in an overflow of the field, the field is truncated at the end by the number of overflow characters.

Format: CHANGE;item;old string;new string

CLCALC

The CLCALC command will zero the CALC registers (stacks) and the 10 memory locations. This command is generally issued prior to beginning a new CALC sequence.

Format: CLCALC

CLEARM

The CLEARM command clears the MASK register for all items. Following this command, the TARGET, RANGE, and SORT functions will consider the entire text content of each item in performing their designated functions.

Format: CLEARM

COPY and NOCOPY

The COPY command sets an internal flag which controls the automatic hardcopy feature. The feature is deactivated by the NOCOPY command and the status of the flag may be obtained using the FLAGS command. CASE/A defaults to the automatic hardcopy turned off. While the feature is turned off, the program will bypass the hardcopy operation following plots and the individual pages of some reports. If the user would prefer a pause after each graphics page for viewing or manual hardcopy, the PAUSE flag may be set using the PAUSE command. This flag is turned off with the NOPAUSE command. The program treats the hardcopy command differently for various device types (TERM;x). In Tektronix mode (TERM;1), the hardcopy device is activated. DEC VT-100 type devices (TERM;2) have no hardcopy capability so a PAUSE is generated instead.

Format: COPY or NOCOPY

DCALC

The DCALC command generates a concise summary of the contents of the CALC registers and the memory locations. The format of the output is as follows:

```
  CALC MEMORY

 1    0.000000
 2    0.000000
 3    0.000000
 4    0.000000
 5    0.000000
 6    0.000000
 7    0.000000
 8    0.000000
 9    0.000000
10    0.000000

  CALC STACKS
 1    0.000000
 2    0.000000
 3    0.000000
 4    0.000000
 5    0.000000
 6    0.000000
 7    0.000000
 8    0.000000
 9    0.000000
10    0.000000
```

Format: DCALC

DELETE

The DELETE command is used to delete a specified record from the data base. The record is marked for deletion and is inaccessible to subsequent FULL commands. The data is not destroyed until the record is actually overwritten. Up to that point the data may be returned to active status with the RESTOR command. For deletion of many records the user may use the DTABLE command. This command marks all records in the active set for deletion and places the data in inactive status.

Format: DELETE;record

DTABLE

The DTABLE command deletes all records in the active set. As with the DELETE function the records are simply flagged as inactive with the information in the records preserved. The data is not destroyed until it is overwritten by new data being entered into the data base. Up to that time the data may be returned to active status with the RESTOR command.

Format: DTABLE

DUAL and NODUAL

The DUAL command sets an internal flag which determines whether the IPU plots will have dual y axes for temperature scales in Fahrenheit and Centigrade. This feature is deactivated by the NODUAL command and the status may be checked by the FLAGS command.

Format: DUAL or NODUAL

FAST

The FAST command is used to make modifications to certain items of every record in the active set. It operates in an identical manner to the MODIFY command except the scope extends from a single record up to the entire active set.

Format: FAST

Example:

```
FAST
ENTER RECORD CONTENT LINES   ! system response
FORMAT--ITEM;VALUE           ! system response
12;NEW TEXT                  ! new data entered
15;34.90
9;10
0                            ! zero to save and exit
```

FLAGS The FLAGS command provides a listing of the current state of the operation control flags. An example of the FLAGS summary follows (these are the system default settings). The commands to set each flag "on" or "off" are provided in the last two columns.

```
 SYSTEM FLAG SUMMARY          |   on      off
                              |
 1 LOGO                  ON   |  LOGO   ,NOLOGO
 2 INVERSE FLAG          OFF  |  INVERSE,NORMAL
 3 COMMAND ECHO          ON   |  ECHO   ,NOECHO
 4 AUTO HARD COPY        ON   |  COPY   ,NOCOPY
 5 PLOT TIME             OFF  |  TIME   ,NOTIME
 6 DUAL Y AXES           OFF  |  DUAL   ,NODUAL
 7 VECTOR GRAPHICS       OFF  |  VECT   ,NOVECT
 8 X AXIS LOG SCALE      OFF  |  LOGX   ,NOLOGX
 9 Y AXIS LOG SCALE      OFF  |  LOGY   ,NOLOGY
10 OVER-WRITE PROTECT    OFF  |  REUSE  ,NOREUSE
11 BOLD PLOT             OFF  |  BOLD   ,NOBOLD
12 PAUSE FOR COPY        OFF  |  PAUSE  ,NOPAUSE
13 NASA LOGO             OFF  |  NASA   ,NONASA
14 COLOR FLAG            OFF  |  COLOR  ,NOCOLOR
13 TASK COMPLETION EST.  OFF  |
13 PLOT LEGEND           ON   |  LEGEND ,NOLEGEND
```

Format: FLAGS

FULL The FULL command loads all active records into the active set. This command is generally issued prior to performing conditional tests on the data base for generation of a specified subset. The user is advised how many records were loaded into the active set upon execution of FULL. Subsequent conditional tests operate on the active set progressively, eliminating unwanted records to result in the desired subset for editing or report operation.

Format: FULL

INVERSE The INVERSE command reverses the function of two other commands: TARGET and RANGE. A summary of the affected commands and their operations follows:

COMMAND: TARGET;item;value
NORMAL: ITEM.EQ.VALUE
INVERSE: ITEM.NE.VALUE

COMMAND: RANGE;item;low;high
NORMAL: ITEM.GT.LOW.AND.ITEM.LT.HIGH
INVERSE: ITEM.LT.LOW.AND.ITEM.GT.HIGH

The status of this flag can be examined using the FLAGS command. The INVERSE function is reset to normal operation using the NORMAL command.

Format: INVERSE

LIST

The `LIST` command produces a listing of the requested record. The item numbers, item titles, and item field contents are listed in a sequential manner. Text fields which extend beyond 32 characters are truncated. Other ways to examine a record's contents include the `PRINT`, `REPORT`, and `EDIT` functions.

Format: `LIST;record number`

LOGO and NOLOGO

The `LOGO` command sets an internal flag which controls whether or not a logo is displayed on IPU plots. The function is deactivated by the `NOLOGO` command. The actual logo displayed depends on the status of the NASA/ flag. If the NASA logo flag is on, the logo will be "NASA". If the NASA logo flag is off, the logo will be " DOUGLAS". The status of all `LOGO` related flags may be checked using the `FLAGS` command.

Format: `LOGO or NOLOGO`

LOGX and LOGY

NOLOGX and NOLOGY

These commands set internal flags to determine whether the x or y axis of subsequent plots will be on a logarithmic scale. The commands may be used in any combination to produce the required results (log-log, semi-log x, semi-log y, etc.). Both flags are off (`NOLOGX` and `NOLOGY`) by default. The status of these flags may be checked using the `FLAGS` command.

Format `LOGX` or `NOLOGX`,

`LOGY` or `NOLOGY`

MODIFY

The `MODIFY` command allows the user to modify several items in a given record. The user is prompted to enter data in an "ITEM;VALUE" format, one per line. Any number of items can be changed in any order. The operation is terminated by a zero item number. The entire record is listed upon completion.

Format: `MODIFY;record number`

Example:

```
MODIFY;1001
2;NEW TEXT
16;40.32
0
```

NORMAL The INVERSE function is reset to normal operation using the NORMAL command. The INVERSE command reverses the function of two other commands: TARGET and RANGE. A summary of the affected commands and their operations follows:

COMMAND: TARGET;item;value
NORMAL: ITEM.EQ.VALUE
INVERSE: ITEM.NE.VALUE

COMMAND: RANGE;item;low;high
NORMAL: ITEM.GT.LOW.AND.ITEM.LT.HIGH
INVERSE: ITEM.LT.LOW.AND.ITEM.GT.HIGH

The status of this flag can be examined using the FLAGS command.

Format: `NORMAL`

OPEN The `OPEN` command allows as many as four data bases to be open at any given time. The user may toggle between these data bases using the `SETDB` command. The data base number parameter is the integer 1, 2, 3, or 4. The data base activated at `IPU` start-up is automatically allocated to data base number 1. The data base name is the name of the data base to be opened. The user may reallocate a data base number to a different data base. In this case the first data base is closed before the new one is opened. To access an alternate data base the user issues the `SETDB` command. The user may then operate on the alternate in the usual manner with all features of the IPU available. The user may return to the original with a `SETDB;1` command.

Format: `OPEN;data base number; data base name`

OUTPUT The `OUTPUT` command will redirect all IPU related output to the specified VAX/VMS file name. The output can be returned to the user's terminal with an `OUTPUT;TERMINAL` command.

Format: `OUTPUT;file name`

PAUSE and NOPAUSE The `PAUSE` command governs the action taken at the completion of a `PLOT` and after each page in schedule reports, etc., when the `NOCOPY` setting is in effect. The command has no effect on operation when the automatic hardcopy feature is in operation (`COPY`). When the automatic hardcopy function is disabled using the `NOCOPY` command, plots will be displayed but the next command will be executed immediately with no pause for viewing the data. This condition is desirable when the final product is being generated by the combination of `PLOT` and subsequent `PLTOVR` commands, and the intermediate `PLOT` product is of no interest. When the `PAUSE` flag is activated, the system will wait for the user to hit a carriage return before proceeding to the next command. During that time, the user may view the data and make manual hardcopies as required. The `PAUSE` feature is deactivated by the `NOPAUSE` command and the status may be checked using `FLAGS`.

Format: `PAUSE or NOPAUSE`

PLOT

The `PLOT` commands generate graphical representation of the data base information in the form of an x-y plot. Four commands exist differing only in the amount of user specified title information allowed. The `PLOT` commands are:

`PLOT0;`N_1`;`N_2`;...`N_n	No plot title, no y axis label
`PLOT1;`N_1`;`N_2`;...`N_n	Y axis label provided on
`y label`	separate line
`PLOT2;`N_1`;`N_2`;...`N_n	Y axis label and title
`y label`	provided on separate lines
`title`	
`PLOT3;`N_1`;`N_2`;...`N_n	Title provided on separate
`title`	line

The first argument, N_1, is the item number of the independent variable (usually time) which constitutes the x axis of the plot. The next arguments represent the item numbers of one or more dependent variables which will be plotted as the y variables. If more than one dependent variable is provided, different plotting symbols will be utilized to allow identification of the different variables, with the variables and symbols correlated in the legend. The data are normally plotted with straight lines connecting the data points with a symbol evenly spaced in 5 or 6 locations to identify the variable. The dependent variable may also be plotted as discrete symbols at every data point by designating the item as a negative number. A mixture of line and symbol plots is allowed. Following the dependent variable list are the optional minimum and maximum y axis ranges. This will override the internally computed y scale. These min and max values must be in floating point form to avoid being confused with an integer dependent variable item number. The user may also provide the minimum and maximum x ranges in a similar manner but the x range may not be specified without the optional y range. The system offers several `FLAGS` settings which affect the way the plots are displayed. These are summarized as follows:

`LOGO,NOLOGO`	Controls the display of a LOGO on the plot
`NASA,NONASA`	Determines the kind of logo displayed (if any)
`TIME,NOTIME`	Controls the display of time and date on the plot
`DUAL,NODUAL`	Dual y axes for Farenheit and Centigrade
`LOGX,NOLOGX`	x axis is logarithmic scale
`LOGY,NOLOGY`	y axis is logarithmic scale
`SETFIT;order`	Data will be plotted as discrete points with a curve fit of specified order placed through points. order=-1 deactivates feature

Format: `PLOTx;indep;dep1[;dep2;dep3;{;ymin;ymax}`
`{;xmin;xmax}]` where x = 0, 1, 2, or 3

PLTOVR

The PLTOVR command allows data from another data base to be overlaid onto an existing plot frame. The typical use is the comparison of transient analytical predictions and test data. A typical command sequence for the PLTOVR function is given in the example at the end of this command description.

The user will generally have two data bases active when performing a PLTOVR. The primary data base will be the data base specified at the time of IPU activation and will be designated as data base #1. The user will open a second data base using the OPEN command to make the second data set available as data base #2. The IPU command echo and automatic hardcopy should be suppressed using the NOECHO and NOCOPY commands.

Prior to beginning the sequence, the user must jump to data base #2 (SETDB;2) and configure the active set to contain the points which will be plotted with the data from data base #1. This may involve the FULL, TARGET, SAMFST, RANGE, etc. commands as required. The user will then return to data base #1 (SETDB;1) and command a PLOT of the desired information from data base #1. Following the creation of the data base #1 plot, the user may command the overlay of x_item, y_item data from data base #2 using PLTOVR (Note: the user does not leave data base #1 to accomplish this). As with the other plot commands, multiple y_items may be specified, and the display of data can be connected points (positive y_item) or discrete data points (negative y_item). The user may generate a hardcopy using the HDCOPY command following the PLTOVR.

Format: `PLTOVR;DB number; x_item1;y_item1 [;y_item2;y_item3..]`

Example:

```
ENTER DATABASE NAME $ ANALYSIS      Database #1 will be ANALYSIS
IPU_ANALYSIS> FULL                  Prepare to plot all data in ANALYSIS
NUMBER OF RECORDS IN TABLE: 500     System response 500 points found
IPU_ANALYSIS> OPEN;2;TESTDATA       Open data base #2 as TESTDATA
IPU_ANALYSIS> SETDB;2               Jump to TESTDATA
NUMBER OF RECORDS IN TABLE: 0       Active set is empty
IPU_ANALYSIS> FULL                  Find all data in TESTDATA
NUMBER OF RECORDS IN TABLE: 10500   System found 10500 points
IPU_ANALYSIS> SAMFST;20             Use only every 20th point
NUMBER OF RECORDS IN TABLE: 527     527 points remain
IPU_ANALYSIS> SETDB;1               Jump back to ANALYSIS
NUMBER OF RECORDS IN TABLE: 500     ANALYSIS active set is still at 500
IPU_ANALYSIS> NOCOPY                Suppress automatic hardcopy
IPU_ANALYSIS> PLOT2;1;16            PLOT ANALYSIS x=item 1, y=item 16
PRESSURE PSIA                       y axis label
TRANSIENT PRESSURE PROFILE          Plot title
[ PLOT DISPLAYED ]

IPU_ANALYSIS> PLTOVR;2;2;-23        Overlay TESTDATA x=item 2,y=item 23 Note: y item is negative so it
                                    will be plotted as discrete points
[ OVERLAY DISPLAYED ]
IPU_ANALYSIS> HDCOPY                Command a hardcopy of the composite plot
```

PRINT

The `PRINT` command generates a headed column printout of the specified items in the active set. The item titles in the heading are taken from the .DDF file for the data base. The number of items is optional but the amount of data to be printed cannot exceed 132 columns. The number of character positions allocated to an item can be set by appending =xx to the item specification in the command. The record number may be printed by specifying a zero item number.

Format: `PRINT;item1 [;item2;item3...;itemn]`

Example: `PRINT;0;12;16;2;5=20`

In the example, the record number (item 0) will be printed in the first column followed by item 12, item 16, item 2, and item 5. The column width for item 5 will be 20 characters wide. If Item 5 is a 60 character alphanumeric field, for example, the data will be automatically word wrapped to multiple lines as required. The system will generate a pause at each full page of information when operating in interactive mode (`TERM;1,2,3,` or `6`). The `PRINT` function may be discontinued at the pause by typing an X instead of only a carriage return. When operating in Batch mode, (`TERM;4`), no pause will occur.

RAMDISK and NORAM

The `RAMDISK` command reduces the I/O time in processing large data bases. The process places the data base information into an in-core array of approximately 500K in size. The user is advised if the data base will not fit. When this feature is active, the user will see a dramatic improvement of system response to commands which are read-only in nature. This includes TARGET, RANGE, PLOTs, and other commands which only read the data base and do not modify its contents. Because the data base is now in-core, the user must be protected against data loss due to unplanned system interruptions or an error termination of the system. To accomplish this, “write” operations to the data base to modify data or to add new data are performed both to the RAMDISK and to the nonvolatile mass storage file. The user will therefore maintain both the RAMDISK and the mass storage disk up to date at all times. The feature is deactivated by the NORAM command.

Format: `RAMDISK`

RANGE

The `RANGE` command tests each record in the active set to determine whether the designated item lies within the specified low value - high value range (inclusive). If the test is positive (within range), the record is retained in the resulting active set. If the test is negative (outside range), the record is eliminated from the active set (but not deleted from the data base). The user is advised how many records passed the conditional test upon completion of the function. The `INVERSE` command reverses the function of the range command so that the new active set contains records where the designated item is outside the specified range.

Format: `RANGE;item;low value;high value`

RECORDS

The `RECORDS` command provides a list of the record numbers in the active set. This command is usually used for diagnostic purposes.

Format: `RECORDS`

RESTORE The RESTORE command is intended to change the status flag for a record or a range of records to indicate the record is active. The command allows two arguments to specify the range of record numbers to be activated. If the second argument is omitted, the single record indicated in the first argument will be activated. The command can be used to reverse the effects of an erroneous DELETE, provided the record in question has not been overwritten by new data added to the data base.

Format: `RESTORE;low record;high record`

REUSE and NOREUSE The REUSE command sets a flag which controls the way CASE/A finds a location for new data. The REUSE feature is off by default. Setting the flag by using the REUSE command results in CASE/A placing new records in the first (lowest record number) inactive record. This inactive record may have contained data at one time which had been deleted. In this case, the old data will be overwritten and lost. If the user turns off the REUSE feature with the NOREUSE command, CASE/A will place new data in the first inactive record which has never been used. The NOREUSE feature will cause the data base to be poorly "packed" but will allow the recovery of old data at any time using the RESTORE command. The user may check the status of the REUSE flag using the FLAGS command.

Format: `REUSE`

RUN The `RUN` command is used to execute a set of IPU commands as a macro. The macro commands are placed in a standard VAX text file using one of the system-resident editors. The `RUN` command will execute all commands in the macro and return control to the user when complete. The macro may also include the DCL command and VAX/VMS commands.

Format: `RUN;file name`

SAMFST The `SAMFST` command will reduce the size of the active set by selecting points on the specified frequency. A `SAMFST;20`, for instance, will select every 20th point in the active set. This is a quick way to reduce the size of large data bases, typical of test data. The routine will always include the first and last point in the active set.

Format: `SAMFST;frequency`

SELECT The `SELECT` command is used to read a data base directly from a file which was created through user defined code. This file must be stored in binary form in a file named `FOR0xx.DAT`. The data base is then renamed according to the data definition file which defines the data storage structure. The `SELECT` command requires the user to specify the record length of the data base in number of words.

Format: `SELECT;logical number; length in words`

Example:
```
ENTER DATABASE NAME: TEST100
SELECT;99;500
TOTAL RECORDS IN TABLE: 120
```

SETDB The SETDB command allows the user to toggle between open data bases. The data base number parameter is the integer 1, 2, or 3. The original data base activated at IPU start-up is automatically allocated to data base number 1.

Format: `SETDB;data base number`

SIGMA The SIGMA command performs a statistical filtering of the data in the active set to remove points which are more than the specified number of standard deviations from the rest of the data points over a particular time step. The routine performs a stepwise scan of the active set in accordance with the specified time step. Time is assumed to be in Item 1 of the data base. For each step a mean and standard deviation are computed for the data points of the specified item. Each step is then re-scanned to eliminate data points which are more than the specified number of standard deviations from the mean over the interval. The function is particularly useful in systematically eliminating noise from recorded test data or flight data.

Format: `SIGMA;item;step;standard_deviation`

Example:

```
IPU_DBNAME> FULL                    Start with FULL set
TOTAL RECORDS IN TABLE: 15001       15001 records found
IPU_DBNAME> SIGMA;12;5.0;3.0          Remove records where Item 12
                                    is more than 3 sigma out over
                                    a 5.0 hr interval
TOTAL RECORDS IN TABLE: 14501
```

SORT The SORT command initiates a numerical (or alphanumerical) sort of the records in the active set. The default is an ascending sort unless the item number is input as a negative number, where upon a descending order sort results. SORT does not actually rearrange the data in the data base. The record numbers in the active set are merely arranged in accordance with the specified sort. PRINT, REPORT, EDIT, and other commands issued following a sort on a given active set will operate on the data base in the sorted order. The SORT command accepts optional item references up to a total of 50. The effect of multiple item references is as follows: The primary sort is performed on the first(leftmost) item. Where a group of the first item have the same value, this group will be sorted by the second item. Where both the first and second item do not change, they will be sorted by the third and so on. SORTs on active sets with greater than 1000 or so entries can take several minutes to complete. A progress report is issued on frequent intervals to advise the user of the current percent complete.

Format: SORT;item [;item2; item3; item4...]

TABLE The TABLE command is used to manually create an active set. The user provides the records to be inserted into the table one per line and terminated by a zero entry. For entry of a large number of records it is recommended that the user place the table command and associated record numbers in a standard VAX text file using the VAX system editor of your choice. Access to VAX DCL and the editor is provided by the DCL command without leaving IPU. Performing the TABLE function in this manner will allow the user to repair errors in the record number entries without retyping the entire string. The text file can be executed as a macro using the RUN command.

Format: `TABLE`

TARGET

The `TARGET` function tests each record in the active set to determine whether the designated item equals the specified value. If the test is positive (equal to value) the record is retained in the resulting active set. If the test is negative (not equal to value) the record is eliminated from the active set (but not deleted from the data base). The user is advised how many records passed the conditional test upon completion of the function. The `INVERSE` command reverses the function of the `TARGET` command so that the new active set contains records where the designated item is not equal to the specified value.

Format: `TARGET;item;value`

TECPRT

The `TECPRT` command generates a headed column printout of the specified items in the active set and is tailored to operate on the Tektronix type device. This routine is similar to the `PRINT` routine which is tailored to VT-100 type devices. The item titles in the heading are taken from the DDF file for the data base. The number of items is optional, but the amount of data to be printed cannot exceed 132 columns. The number of character positions allocated to an item can be set by appending =xx to the item specification in the command. The record number may be printed by specifying a zero item number.

Format: `TECPRT;item1[;item2]`

Example: `TECPRT;0;12;16;2;5=20`

In the example, the record number (item 0) will be printed in the first column followed by item 12, item 16, item 2, and item 5. The column width for item 5 will be 20 characters wide. If Item 5 is a 60-character alphanumeric field, for example, the data will be automatically word wrapped to multiple lines as required. The system will generate a pause at each full page of information when the automatic hardcopy is turned off. The routine requires the terminal type set to 1 (`TERM;1`).

TERM

The `TERM` command sets the system terminal type to permit the proper interpretation of editing key inputs and graphics output. A summary of terminal types currently supported follows:

`TERM;1` Tektronix 4014

`TERM;2` VT-100 and VT-200

`TERM;3` DUAL VT100/4014

Format: `TERM;i`

TIME and NOTIME

The `TIME` command determines whether the time and date will appear on the subsequent plots. The default is the `NOTIME` state which does not generate time and date output. The status of the `TIME` flag may be verified using `FLAGS`.

Format: `TIME` or `NOTIME`

APPENDIX C

CASE/A INSTALLATION NOTES

This section provides instructions for installing the CASE/A software onto a Digital Equipment Corporation© VAX computing system. The software is provided on 6250 density tapes. The CASE/A system is developed to run under the VAX VMS environment and uses the VT100 and TEK4014 display systems.

The installation process requires two steps:

1) **System Manager installation of CASE/A code and data base:**
This installation is done by the Systems Manager usually from a tape copy onto the VAX system. The Systems Manager will have the proper access rights to create the directories and install the CASE/A code and data base.

2) **User setup and customization:**
After the Systems Manager completes the installation in part 1, OR, if CASE/A is already installed on your system, the user should follow these instructions to set up the CASE/A User directory and assignments in the LOGIN.COM file to create the necessary environment for learning CASE/A and developing new simulation cases.

System Manager installation of CASE/A code and data base:

The CASE/A system can be copied from the 6,250 tapes provided. To install CASE/A from tape, the System Manager at the user site should use the VMS BACKUP facility as follows:

```
$ BACKUP TAPE$:SAVESET1.BAK [*...]/OWNER=USERID
```

This will backup the CASE/A code and support files to the default directory. If the installation is new then the data base directory will also need to be created as follows:

```
$ BACKUP TAPE$:SAVESET2.BAK [*...]/OWNER=USERID
```

Here, `TAPE$` refers to the tape drive logical name and `USERID` refers to the user who will own the files and be responsible for maintaining the CASE/A system. This procedure will create the parent directory [CASEA] and the subdirectories [.CODE], [.USER], and [.DATA] (if the data base is backed up).

OPTIONAL: The `LIBCREATE.COM` file is used to create the FORTRAN code library. This command file should be run only if the CASE/A programmers/users wish to use the VAX Debugger to debug model cases. The FORTRAN files are then compiled with the Debug option. Otherwise, this command file need not be run. The `LIBCREATE.COM` file is shown in figure C-4. The FORTRAN files that are compiled by this routine and then loaded into the CASE/A library are contained in `LIB1LIST.COM, LIB2LIST.COM, LIB3LIST.COM, LIB4LIST.COM,` and `LIB5LIST.COM` which are shown in figures C-5, C-6, C-7, C-8, and C-9, respectively. The files in LIB5LIST.COM must be compiled without the CHECK=BOUNDS option.

To install the CASE/A system and enable the VAX Debugger, type the following:

```
$ @LIBCREATE
```

The following are messages displayed on the screen by the command file:

```
CASEA LIB1 COMPILATION INITIATED....              ! System responses
CASEA LIB1 COMPILATION COMPLETED.
CASEA LIB2 COMPILATION INITIATED....
CASEA LIB2 COMPILATION COMPLETED.
CASEA LIB3 COMPILATION INITIATED....
CASEA LIB3 COMPILATION COMPLETED.
CASEA LIB4 COMPILATION INITIATED....
CASEA LIB4 COMPILATION COMPLETED.
CASEA LIB5 COMPILATION INITIATED....
CASEA LIB5 COMPILATION COMPLETED.
CASEA LIB1 ROUTINE LOADING INITIATED....
CASEA LIB1 ROUTINES LOADED INTO LIBRARY.
CASEA LIB2 ROUTINE LOADING INITIATED....
CASEA LIB2 ROUTINES LOADED INTO LIBRARY.
CASEA LIB3 ROUTINE LOADING INITIATED....
CASEA LIB3 ROUTINES LOADED INTO LIBRARY.
CASEA LIB4 ROUTINE LOADING INITIATED....
CASEA LIB4 ROUTINES LOADED INTO LIBRARY.
CASEA LIB5 ROUTINE LOADING INITIATED....
CASEA LIB5 ROUTINES LOADED INTO LIBRARY.
CASEA LIBRARY CREATION COMPLETED SUCCESSFULLY.
```

User Set-up and Customization:

After the CASEA.CODE, DATA, USER directories are available to the user, the following steps should be followed to set up the CASE/A environment and enable the user to get underway.

1) Create a directory under your user (user root name) directory as follows:

CREATE /DIRECTORY [**userid.**CASEA]
where **userid** is your user root name directory

2) Copy all files from the DEVICE:[CASEA.USER] directory to the directory created in step 1 as follows. This will move example models, command files, and archive models to your directory.

COPY **DEVICE:**[CASEA.USER]*.* **USER$:**[**userid**.CASEA]
where **DEVICE** is the disk drive logical name where the CASE/A code was installed by the System Manager.
where **USER$** is the logical device on which your user root name directory resides.
where **userid** is your user root name directory.

3) Next copy all archive files from the CASEA.USER.ARCHIVE directory to install example models as follows:

COPY **DEVICE:**[CASEA.USER.ARCH]*.* **USER$:**[**userid**.CASEA.ARCH]
where **DEVICE** is the disk drive logical name where the CASE/A code was installed by the System Manager.
where **USER$** is the logical device on which your user root name directory resides.
where **userid** is your user root name directory.

4) Add the statements shown in figure C-2 to your LOGIN.COM file. Remember to run @LOGIN or logout and log back in to activate these assignments. An explanation of these statements is provided below.

5) Now refer to the Introductory Example in Section 1.3 and the program initiation information in Section 2.1, especially if you are a new user. Example 1: CASE ATC in Section 1.3 will provide instruction as to how to link the CASE/A code in your [userid.CASEA] directory and run this example to get you started with the CASE/A tool.

File Structure

The System Manager and User installation steps above create the CASE/A system file structure illustrated in figure C-1. The CASE/A system is maintained in three VAX directories as shown in figure C-1. The FORTRAN code for the component routines and other CASE/A routines is maintained in the `[CASEA.CODE]` directory. The component data base files (including component default data) reside in the `[CASEA.DATA]` directory. The logical names `CASEA$CODE` and `CASEA$DATA` serve as pointers to these two directories. Since it is mandatory that these pointers be active during all interaction with the CASE/A program, it is highly recommended that they be created in the user's `LOGIN.COM` file as shown in figure C-2.

The last program directory is the user directory `[userid.CASEA]`. This is the directory from which the user will build and execute CASE/A models. It is advisable for each user to have a customized working directory.

The `LINKCAS.COM` file (figure C-3) is used to link with the CASE/A program and create an executable image from which to run a model case. Each user will have a copy of one or more CASE/A executable modules in the [userid.CASEA] directory. To invoke LINKCAS.COM type the following:

```
@CASEA$CODE:LINKCAS
```

Also ensure that the default directory is set to the location where the executable image is to be written (usually the [userid.CASEA] directory).

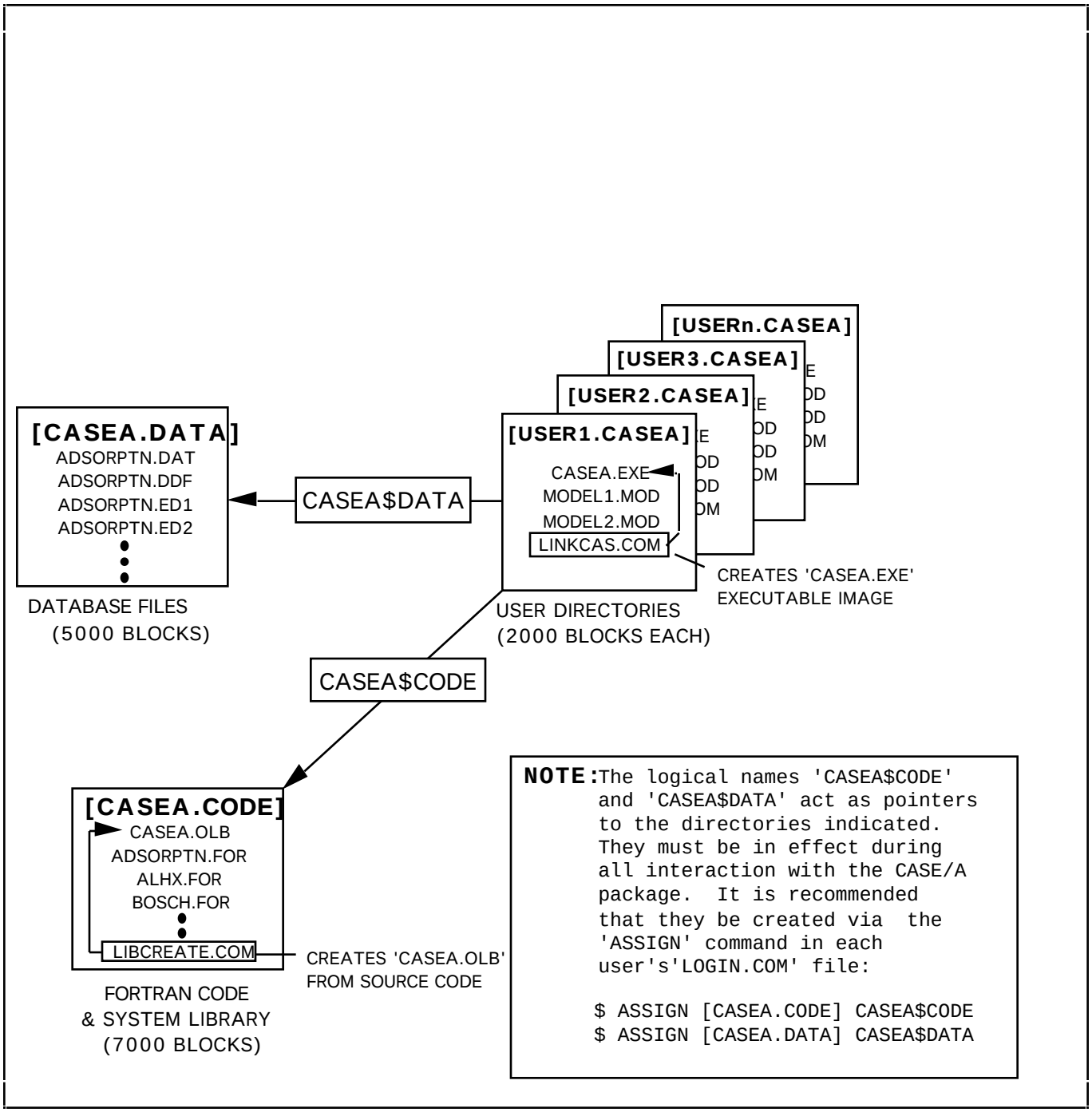


Figure C-1. CASE/A system file architecture.

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                !
! THE FOLLOWING CODE SEGMENT SHOULD BE INCLUDED  !
! IN THE USER'S LOGIN.COM FILE TO ASSURE PROPER  !
! OPERATION OF THE CASEA SYSTEM.                 !
!                                                !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!
$ CASEADIR=="SET DEF DEVICE:[USERNAME.CASEA]"
$ DATA =="SET DEF DEVICE:[CASEA.DATA]"
$ CODE =="SET DEF DEVICE:[CASEA.CODE]"
$ ASSIGN DEVICE:[CASEA.CODE] CASEA$CODE
$ ASSIGN DEVICE:[CASEA.DATA] CASEA$DATA
$ SET TERM/NOWRAP
$ SET TERM/ESCAPE
$ EXIT
!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!                                                !
! THE DEVICE NAME AND USERNAME MUST BE TAILORED  !
! TO THE USER'S VAX COMPUTER SYSTEM AS FOLLOWS:  !
!                                                !
! DEVICE = DISK DRIVE LOGICAL NAME (EX: $DISK1) !
! USERNAME = USER'S ROOT DIRECTORY NAME          !
!                                                !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Figure C-2. Recommended LOGIN.COM file segment.

```
$ SET VERIFY
$ !
$ ! THIS COMMAND PROCEDURE CONSTRUCTS THE CASE/A EXECUTABLE IMAGE
$ !
$ ! Usage:    @LINKCAS NAME
$ !
$ ! where:    NAME = The name of the object file of the user-developed OPS
$ !                  Fortran logic.  The recommended naming convention for the
$ !                  OPS logic file is to use the model name to which the
$ !                  logic applies.  If NAME is not specified, the name
$ !                  'CASEA' will be used with the default 'OPS.FOR' file
$ !                  which contains dummy user operations routines.
$
$ !  WARNING:  This command procedure assumes that the assignment for the
$ !            logical 'CASEA$CODE' has been made prior to invoking
$ !            it.  The assignment statement is listed below:
$
$ !            ASSIGN Device2:[Directory2] CASEA$CODE
$
$ !            The PLOT10 object libraries must be located on
$ !            'Device1' in 'Directory1'.  The CASEA system library must
$ !            be located on 'Device2' in 'Directory2'.  The OPS logic
$ !            file must be located on the same directory as this
$ !            command file.  The user may wish to set up their own
$ !            logicals or insert the actual directory names in the LINK
$ !            statements below.
$
$
$ if "''P9'" .eqs. "" then goto CASEA_8
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,'P4,'P5,'P6,'P7,'P8,'P9,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "''P1'"
$
$ goto end
$
$ CASEA_8:
$ if "''P8'" .eqs. "" then goto CASEA_7
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,'P4,'P5,'P6,'P7,'P8,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "''P1'"
$
$ goto end
$
```

Figure C-3. LINKCAS.COM file.

```
$ CASEA_7:
$ if "''P7'" .eqs. "" then goto CASEA_6
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,'P4,'P5,'P6,'P7,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "''P1'"
$
$ goto end
$
$ CASEA_6:
$ if "''P6'" .eqs. "" then goto CASEA_5
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,'P4,'P5,'P6,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "''P1'"
$
$ goto end
$
$ CASEA_5:
$ if "''P5'" .eqs. "" then goto CASEA_4
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,'P4,'P5,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "''P1'"
$
$ goto end
$
$ CASEA_4:
$ if "''P4'" .eqs. "" then goto CASEA_3
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,'P4,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "''P1'"
$
$ goto end
$
```

Figure C-3. LINKCAS.COM file (continued).

```
$ CASEA_3:
$ if "'P3'" .eqs. "" then goto CASEA_2
$
$ LINK/DEB/EXEC='P1 'P1,'P2,'P3,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "'P1'"
$
$ goto end
$
$ CASEA_2:
$ if "'P2'" .eqs. "" then goto CASEA_1
$
$ LINK/DEB/EXEC='P1 'P1,'P2,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "'P1'"
$
$ goto end
$
$ CASEA_1:
$ if "'P1'" .eqs. "" then goto casea
$
$ LINK/DEB/EXEC='P1 'P1,-
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: " + "'P1'"
$
$ goto end
$
```

Figure C-3. LINKCAS.COM file (continued).

```
$ casea:
$ LINK/DEB/EXEC=CASEA  -
  CASEA$CODE:CASEAMAIN,CASEA/LIB,-
  SYS$INPUT:/OPTIONS
  ISD_MAX=150
$ ON ERROR THEN GOTO ERR
$
$ write sys$output "EXECUTABLE IMAGE CREATED: CASEA"
$
$ goto end
$
$ err:
$ write sys$output "LINKER ERROR IN CREATION OF EXECUTABLE..."
$ write sys$output "EXECUTABLE IMAGE CREATION ABORTED."
$
$ goto end
$
$ end:
$ SET NOVERIFY
$ exit
```

Figure C-3. LINKCAS.COM file (continued).

```
$ !
$ ! Compile all LIB1LIST Fortran files
$ !
$ FOR1:
$ WRITE SYS$OUTPUT "CASEA LIB1 COMPILATION INITIATED...."
$ FOR/DEB/NOOP/CON=99/CHECK=BOUNDS CASEAMAIN
$ FOR/DEB/NOOP/CON=99/CHECK=BOUNDS @LIB1LIST
$ ON ERROR THEN GOTO ERR1
$ WRITE SYS$OUTPUT "CASEA LIB1 COMPILATION COMPLETED."
$ !
$ ! Compile all LIB2LIST Fortran files
$ !
$ FOR2:
$ WRITE SYS$OUTPUT "CASEA LIB2 COMPILATION INITIATED...."
$ FOR/DEB/NOOP/CON=99/CHECK=BOUNDS @LIB2LIST
$ ON ERROR THEN GOTO ERR1
$ WRITE SYS$OUTPUT "CASEA LIB2 COMPILATION COMPLETED."
$ !
$ ! Compile all LIB3LIST Fortran files
$ !
$ FOR3:
$ WRITE SYS$OUTPUT "CASEA LIB3 COMPILATION INITIATED...."
$ FOR/DEB/NOOP/CON=99/CHECK=BOUNDS @LIB3LIST
$ ON ERROR THEN GOTO ERR1
$ WRITE SYS$OUTPUT "CASEA LIB3 COMPILATION COMPLETED."
$ !
$ ! Compile all LIB4LIST Fortran files
$ !
$ FOR4:
$ WRITE SYS$OUTPUT "CASEA LIB4 COMPILATION INITIATED...."
$ FOR/DEB/NOOP/CON=99/CHECK=BOUNDS @LIB4LIST
$ ON ERROR THEN GOTO ERR1
$ WRITE SYS$OUTPUT "CASEA LIB4 COMPILATION COMPLETED."
$ !
$ ! Compile all LIB5LIST Fortran files
$ !
$ FOR5:
$ WRITE SYS$OUTPUT "CASEA LIB5 COMPILATION INITIATED...."
$ FOR/DEB/NOOP/CON=99 @LIB5LIST
$ ON ERROR THEN GOTO ERR1
$ WRITE SYS$OUTPUT "CASEA LIB5 COMPILATION COMPLETED."
$ !
$ !
$ !
$ ! Load all LIB1LIST Object files into CASEA library
$ !
$ LIB1:
$ LIBRARY/CREATE CASEA
$ WRITE SYS$OUTPUT "CASEA LIB1 ROUTINE LOADING INITIATED..."
$ LIB/REPLACE CASEA @LIB1LIST
$ ON ERROR THEN GOTO ERR2
$ WRITE SYS$OUTPUT "CASEA LIB1 ROUTINES LOADED INTO LIBRARY."
```

Figure C-4. LIBCREATE.COM file.

```
$ !
$ ! Load all LIB2LIST Object files into CASEA library
$ !
$ LIB2:
$ WRITE SYS$OUTPUT "CASEA LIB2 ROUTINE LOADING INITIATED..."
$ LIB/REPLACE CASEA @LIB2LIST
$ ON ERROR THEN GOTO ERR2
$ WRITE SYS$OUTPUT "CASEA LIB2 ROUTINES LOADED INTO LIBRARY."
$ !
$ ! Load all LIB3LIST Object files into CASEA library
$ !
$ LIB3:
$ WRITE SYS$OUTPUT "CASEA LIB3 ROUTINE LOADING INITIATED..."
$ LIB/REPLACE CASEA @LIB3LIST
$ ON ERROR THEN GOTO ERR2
$ WRITE SYS$OUTPUT "CASEA LIB3 ROUTINES LOADED INTO LIBRARY."
$ !
$ ! Load all LIB4LIST Object files into CASEA library
$ !
$ LIB4:
$ WRITE SYS$OUTPUT "CASEA LIB4 ROUTINE LOADING INITIATED..."
$ LIB/REPLACE CASEA @LIB4LIST
$ ON ERROR THEN GOTO ERR2
$ WRITE SYS$OUTPUT "CASEA LIB4 ROUTINES LOADED INTO LIBRARY."
$ !
$ ! Load all LIB4LIST Object files into CASEA library
$ !
$ LIB5:
$ WRITE SYS$OUTPUT "CASEA LIB5 ROUTINE LOADING INITIATED..."
$ LIB/REPLACE CASEA @LIB5LIST
$ ON ERROR THEN GOTO ERR2
$ WRITE SYS$OUTPUT "CASEA LI5 ROUTINES LOADED INTO LIBRARY."
$ !
$ !
$ WRITE SYS$OUTPUT "CASEA LIBRARY CREATION COMPLETED SUCCESSFULLY."
$ GOTO END
$ !
$ !
$ ERR1:
$ WRITE SYS$OUTPUT "CASEA FORTRAN COMPILATION ERROR."
$ WRITE SYS$OUTPUT "CASEA LIBRARY CREATION ABORTED."
$ GOTO END
$ !
$ !
$ ERR2:
$ WRITE SYS$OUTPUT "CASEA ROUTINE LOADING ERROR."
$ WRITE SYS$OUTPUT "CASEA LIBRARY CREATION ABORTED."
$ GOTO END
$ !
$ !
$ END:
$ EXIT
```

Figure C-4. LIBCREATE.COM file (continued).

```
ADEIN,          ADEOUT,         ADSORPTN,       AFSPE,          -
AFSPEHX,        ALARM,          ANSWER,         ARCHIVE,        -
ARCHFILE,       ASCII_TEC,      ASSIGN,         AVRAGE,         -
AUTOPLOT,                                                       -
BCK_GD_CLR,     BDN,            BDNSHT,                         -
BEDLOAD_DEFLOW, BENCH,          BIVAR,          BLANK,          -
BLDREC,         BLF,            BLFSHT,         BLOCK,          -
ADEIN,          ADEOUT,         ADSORPTN,       AFSPE,          -
AFSPEHX,        ALARM,          ANSWER,         ARCHIVE,        -
ARCHFILE,       ASCII_TEC,      ASSIGN,         AVRAGE,         -
AUTOPLOT,                                                       -
BCK_GD_CLR,     BDN,            BDNSHT,                         -
BEDLOAD_DEFLOW, BENCH,          BIVAR,          BLANK,          -
BLDREC,         BLF,            BLFSHT,         BLOCK,          -
BMR,            BOSCH,          BRT,            BRTSHT,         -
BUP,            BUPSHT,         B_CHX,          CABIN,          -
CALC,           CAP,            CASEREAD,       CASEWRIT,       -
CDCODE,         CDEL,           CDELA,          CEDIT,          -
CFIELD,         CFR,            CHANGE,         CHKITM,         -
CHKREC,         CHX,            CILLCHAR,       CIRCLE,         -
CLCALC,         CLDRAI,         CLDRIA,         CLOADCASE,      -
CLONE,          CLOSTS,         CLS,            CLS2,           -
CMPOPEN,        CNETWK,         CNEWCASE,       CNHX,           -
CNTRLLR,        COLOR,          COMPDP,         COMPUTE,        -
CONCALC,        CONDP,          CONECT,         CONINIT,        -
CONV,           COPY,           COPYA,          CORRECT,        -
CP,             COPYALL,        CREADAL,        CREADALC,       -
CREADFL,        CREADIN,        CREW,           CTOLOWERC,      -
CTOUPPERC,      CURSOR,         CUTALL,         D1POLY,         -
DAXPY,          DBIVAR,         DBOPEN,         DCALC,          -
DCLFOR,         DCONV,          DDOT,           DEFLOW,         -
DEHUM,          DEL,            DELCAS,         DELCN,          -
DELEQ,          DELLAB,         DELNOTE,        DELREC,         -
DENVIS,         DEWPT,          DGEFA,          DGESL,          -
DINTER,         DIR,            DIR_ADV,        DIST
```

Figure C-5. LIB1LIST.COM file.

```
DMNETWK,        DPCS,           DPOFA,          DPOSL,          -
DRAWC,          DRBIVAR,        DRLABL,         DRNOTE,         -
DTABLE,         DTNETWK2,       DUPLICATE,      EDC,            -
EDITOR,         EDT,            ENCODE,         ENTER,          -
ENTRY,          EQLOAD,         EQOPEN,         EQSOLVE,        -
EQWRIT,         ERRDUMP,        ETA_DEFLOW,     EVAP,           -
EXPORT,         FAST,           FCELL,          FDUMP,          -
FIELD,          FILE,           FILREC,         FILTER,         -
FIND,           FINDC,          FINDRM,         FLAG,           -
FLOLEG,         FMPOPEN,        FOODPROC,       FORMAT,         -
FRAME1,         FRICTDP,        GETC,           GETDAT,         -
GETDATNEW,      GETI,           GETK,                           -
GETP,           GETPP,          GETT,           GETU,           -
GFINIT,         GFTINIT,        GIMAG,                          -
GRALPH,         GRAPH,          GRBELL,         GRBLANK,        -
GRBUFR,         GRCHRZ,         GRCOPY,         GRCUSR,         -
GRDRAW,         GRDRAWD,        GREFRM,         GRERAS,         -
GRFFRM,         GRFNTT,         GRHOME,         GRINIT,         -
GRLBAB,         GRLBCT,         GRMOVE,         GRRSET,         -
GRSCREEN,       GRSEE,          GRSMBL,         GRTERM,         -
GRVECT,         GRWAIT,         H2OSEP,         HATCH,          -
HCONVECT,                                                       -
HEADER,         HEATER,         HELP,           HIT,            -
HITBOX,         HX,             IMPORT,         INPFRAC,        -
INTER,          IONEXCH,        ISAMAX,                         -
ISOLCO2,        ISOLH2O,        ISOPCO2,        ISOPH2O,        -
ISTAT,          ITEMIO,         ITMOUT,         JOIN,           -
KAM2AS,         KAS2AM
```

Figure C-6 LIB2LIST.COM file.

```
KBIT,                                                                   -
KHECK,          KHECKA,             LABL,                   LBIT,       -
LINCHK,         LIOH,               LISOPEN,                LIST,       -
LISTAB,         LIST_EDITOR,        LIST_FULL_PAGE_MANAGER, LIST_MANAGER,-
LIST_NEW,       LIST_REGESTER,      LOADCOND,               LOCATE,     -
LOGOCIR,        LOGO_MDAC,          LOGO_NASA,              LOGO_SVT,   -
LOOPWARN,       LSQFIT,             LTSEMCIR,               MASKER,     -
MASSFRAC,       MERGE,              MERGE_IN,               MERGE_OUT,  -
MNETWK,         MODBAK,             MODULE,                 MOLEFRAC,   -
MOLSIEV,        MOVALL,             MOVEIT,                             -
MOVLAB,         MOVNOTE,            MSPLT,                  NCHAR,      -
NODE,           NOTE,               NUM_STEP_DEFLOW,        O2N2,       -
OPENDB_X,       OPENTS,             OPS,                                -
PASSIVE,        PDEL,               PINIT,                              -
PIPE,           PIPEDP,             PLANT,                  PLTDATA,    -
PLTFILE,        POINTCON,           POLYSM,                 PRELIM,     -
PREPORT,        PREPRO,             PREWAST,                PRHEAD,     -
PRIMDB,         PRNTSS,             PROPS,                  PSATW,      -
PSEUDO,         PSPCFD,             PSPEC,                  PTMOD,      -
PULL,           PULLSTX,            PUMP,                   PUSH,       -
PUSHSTX,        PWVSUM,             QEXCHG,                 QLOAD,      -
RACK,           RAD,                RAMDISK,                RANDIN,     -
RANDOU,         RANGE,                                      RBIVAR,     -
RBYTE,          READAL,             REDRAW,                 REDUCE,     -
RENEQ,          REPORT,             RESET1,                 RESTOR,     -
RETRIEVE,       RHEAD,              RINTER,                             -
RJUSTIFY,       RNSS,               RO,                     ROTATE,     -
RTREVE,         RTSEMCIR,           SABAT,                  SABCHX,     -
SABDEL,         SABGCAL6,           SABH2O,                 SABRAT,     -
SAMFST
```

Figure C-7. LIB3LIST.COM file.

```
SAMPLE,          SATPR,          SAVE,             SAVEAS,          -
SAWD,            SAWD_ETA,       SBYTE,            SCALE,           -
SCALER,          SCREEN_DUMP,    SCREEN_MANAGER,   SCREEN_MANUNX,  .-
SCREEN_MANVAX,   SECURE,         SEMIRECT,         SEQUENCE,        -
SETC,            SETFLO,         SETK,             SETP,            -
SETPRIM,         SETT,           SETU,             SFORMAT,         -
SFWE,            SFWET,          SINK,             SMVBITS,         -
SOLVE,           SORTIEQ,        SORTR,            SOURCE,          -
SPE1,            SPE2,           SSCAL,            SSOUT,           -
STAMP,           STATS,          SUB1,             SUB2,            -
SUB3,            SUBENTRY,       SUBSYS,           SUBTTL,          -
SUB_CONSTIT,     SUB_EDITOR,     SUB_MANAGER,      SUM,             -
SUMINIT,         SUMMARY,        SUMXN,            SUMXNY,          -
SYSBAL,          SYSCLK,         TABLE,            TALLY            -
TANK,            TARGET,         TBOUND,           TBUS,            -
TDBAR,           TDEL,           TDELA,            TDGRID,          -
TDXAXIS,         TDXAXISLOG,     TDYAXIS,          TDYAXISLOG,      -
TEC,             TECREPORT,      TEK_ADV,          TEK_HOM,         -
TEMPNET,         TEST,           TICS,             TIMER,           -
TIMESC,          TIMESTEP,       TNETWK,           TNETWK2,         -
TOGGLE_SCREEN,   TRANSBOD,       TRANSCON,         TRANSLT,         -
TRANSLT1,        TREDUCE,        TSPCFD,           TSTEP,           -
TSTORE,          TYCON,          UNASSIGN,                          -
UPDPLT,          USERPROPS,      VALVE,            VCD,             -
VISC,            VTFIELD,        VTINVERT,         VTLBAB,          -
VTNORMAL,        WARN,           WASH,             WAS_I_MOD,       -
WQM,             WRITCON,        XMIT1,            XMIT2,           -
XMIT3,           XYCODE
```

Figure C-8. LIB4LIST.COM file.

```
ELIMINAT,       EXCHAN,          FIT,          GRLBAB,       -
INTERSEC,       PLTOVR,          POINT,        RANK,         -
SCAN,           SCREDT,          SELECT,       SIGMA,        -
SORT,           TDPLOT,          UNION
```

Figure C-8 LIB5LIST.COM file.